Appendices
A: Co-ordinate Systems

A:1 Referring to voxels by position

**Real co-ordinates, \(x \in \Xi\)**

\(\Xi\) is the subset of \(\mathbb{R}^3\) which is imaged. \(\Xi\) is partitioned into \(K\) voxels, \(\zeta = \{V_k\}_{k=1}^K\) \(V_k \subset \Xi,\ k = 1,\ldots,K;\) \(V_k \cap V_{k'} = \emptyset\) for \(k \neq k'\); and \(\bigcup_{k=1}^K V_k = \Xi\).

Sometimes it is convenient to refer to a voxel in an image using Cartesian coordinates. For an image \(Y = (Y_1,\ldots,Y_K)\), abusing the notation somewhat, let \(Y(x)\) be the value of the voxel containing point \(x\):

\[
Y(x) = \begin{cases} 
\sum_{k=1}^K Y_k \{x \in V_k\} & \text{for } x \in \Xi \\
0 & \text{for } x \notin \Xi 
\end{cases}
\]

where \(\Xi = \mathbb{R}^3 \setminus \Xi\) the compliment of \(\Xi\) in \(\mathbb{R}^3\), and a logical expression in brackets “\{\text{•}\}” takes the value one if the argument is true and zero otherwise, as advocated by Knuth (1992).

**Voxel Co-ordinates**

If the voxels are identical in shape and size, and are regularly arranged, then the \(D\)-dimensional image is conveniently stored in a \(D\)-dimensional array. In this case, voxels are most conveniently referred to by their array indices. Usually the image is stored in an array such that increases in the row, column, and plane indices correspond to increases in the \(X\), \(Y\), and \(Z\) directions respectively.

This co-ordinate system is referred to as **voxel co-ordinates**, since the position of each voxel is specified by the displacement in each axial dimension measured in whole numbers of voxels from a given origin. For the standard orientation of the co-ordinate axes in PET, the most left-posterior-lower voxel in the image space is (1,1,1) (fig.92).

![Figure 92](image-url)

Voxel co-ordinates of voxels (pixels) at the left-lower of a two dimensional image.
Suppose that $\Xi$ is cuboid, oriented parallel to the axes, and partitioned into $K = \text{xdim} \times \text{ydim} \times \text{zdim}$ cuboid voxels. Here $\text{xdim}$ is the width (x-dimension) of $\Xi$, measured in voxels. If the centre of the left-posterior-lower voxel of $\Xi$ is at $(x_o, y_o, z_o)$, and voxels are of uniform dimensions $h_x \times h_y \times h_z$ then the voxel co-ordinates $x^v = (x^v, y^v, z^v)$ are related to the real co-ordinates $x$ by:

$$x = (x, y, z) = (x^vh_x + x_o, y^vh_y + y_o, z^vh_z + z_o)$$

for $x^v \in [-0.5, \text{xdim} + 0.5] \times [-0.5, \text{ydim} + 0.5] \times [-0.5, \text{zdim} + 0.5]$

$$x^v = ((x-x_o)/h_x, (y-y_o)/h_y, (z-z_o)/h_z)$$

for $x \in \Xi$

Denote voxel co-ordinates using brackets thus: $Y_{[x^v]} = Y_{(x)}$, where $x$ and $x^v$ are related as above.

**A:2 Tri-linear interpolation**

Recall that the reconstructed images $\hat{\lambda}$ are estimates of $\lambda$, itself a step function approximating the continuous function $\lambda(x)$. Thus, an image $Y$ derived from $\hat{\lambda}$ can be regarded as a step function approximating an underlying continuous function $Y(x)$. In many cases it is necessary to obtain estimates of $Y(x)$ for arbitrary locations, for example when re-sampling an image after a change of co-ordinate axes. In these situations some form of interpolation of the image vector $Y$ is desirable.

If we view the image $Y$ as approximating $Y(x)$ at the centres of the voxels, then for locations $x$ other than the voxel centres, an estimate $\tilde{Y}(x)$ of $Y(x)$ can be obtained by interpolating between the values at neighbouring voxel locations. Tri-linear interpolation is usually employed in PET image analysis. This is the simplest form of interpolation for three-dimensional data. The interpolated value at a given point is a linear combination of the values at eight neighbouring voxels whose centres define the cuboid containing the point.

Let $x^v = (x^v, y^v, z^v)$ be $x$ in voxel co-ordinates, and let $t = (t_x, t_y, t_z) = x^v - (\lfloor x^v \rfloor, \lfloor y^v \rfloor, \lfloor z^v \rfloor)$ for $\lfloor \bullet \rfloor$ the floor function, rounding the argument towards minus infinity. Then, for $\lceil \bullet \rceil$ the ceil function, rounding towards plus infinity:

$$\tilde{Y}(x) =$$

$$(1-t_x)(1-t_y)(1-t_z) Y_{[\lfloor x^v \rfloor, \lfloor y^v \rfloor, \lfloor z^v \rfloor]} + (1-t_x)(1-t_y)(t_z) Y_{[\lfloor x^v \rfloor, \lfloor y^v \rfloor, \lceil z^v \rceil]}$$

$$+ (1-t_x)(t_y)(1-t_z) Y_{[\lfloor x^v \rfloor, \lceil y^v \rceil, \lfloor z^v \rfloor]} + (1-t_x)(t_y)(t_z) Y_{[\lfloor x^v \rfloor, \lceil y^v \rceil, \lceil z^v \rceil]}$$

$$+ (t_x)(1-t_y)(1-t_z) Y_{[\lceil x^v \rceil, \lfloor y^v \rfloor, \lfloor z^v \rfloor]} + (t_x)(1-t_y)(t_z) Y_{[\lceil x^v \rceil, \lfloor y^v \rfloor, \lceil z^v \rceil]}$$

$$+ (t_x)(t_y)(1-t_z) Y_{[\lceil x^v \rceil, \lceil y^v \rceil, \lfloor z^v \rfloor]} + (t_x)(t_y)(t_z) Y_{[\lceil x^v \rceil, \lceil y^v \rceil, \lceil z^v \rceil]}$$

Clearly interpolation of $Y$ introduces some smoothing.
B: Smoothing Convolution

B:1 Smoothing convolution

For a continuous function $Y(x)$, $x \in \mathbb{R}^D$, smoothing is achieved by convolving the function with a filter kernel $f(x)$, to obtain a new function $Y \ast f(x)$:

$$X \ast f(x) = \int f(r) Y(x+r)dr \quad (76)$$

Here integration is over the whole range of $x$. The filter kernel, $f(x)$, satisfies:

$$\int f(x)dx = 1$$

The kernel is a continuous function, usually with a single local maximum at the origin, and with value decreasing as $x$ becomes distant from 0.

B:2 Moving average filter

The discrete analogue of convolution is that of a moving average filter. Although technically incorrect, smoothing of images is frequently described as “convolution with a kernel”. Suppose our discretisation of the image space $\Xi \subset \mathbb{R}^D$ is of $K$ voxels $\mathcal{V} = \{V_k: k = 1, \ldots, K\}$, and let $x_k$ be the centre of voxel $k$. Then for an image of voxel values, $Y = \{Y_1, \ldots, Y_K\}$, the smoothed version of this is $Y^S = \{Y^S_1, \ldots, Y^S_K\}$, given by:

$$Y^S_k = \sum_{k'=1}^{K} f(x_{k'} - x_k) Y_{k'} \quad (77)$$

For a particular voxel, the smoothed image is obtained by positioning the filter kernel on the centre of the voxel, evaluating it on the lattice of points corresponding to the centres of the voxels to obtain the weightings for the voxels, and then summing the weighted voxel values.

Regular discretisation, constant weights, moving average

If the voxels are identical in size and shape, and are regularly arranged, then the set of weights $(f(x_{k'} - x_k))$ for any voxel $k$ will be identical (ignoring boundary effects). In this case the weights can be computed in advance, giving an image of the filter kernel. The weights can then be explicitly normalised to sum to unity, and the smoothing is a simple moving average.

B:3 Edge effects & boundary truncation smoothing

For voxels $k$ close to the edge of the image space, the filter kernel when located at these voxels will have positive values outside the image space. I.e. $f(x-x_k)$ is non negative for some $x \in \overline{\Xi} = \mathbb{R}^D \setminus \Xi$. For such a voxel $k$, the sum of the weights $f(x_{k'} - x_k)$ over all voxels $k'$ is less than one. The remaining “weight” of the smoothing kernel corresponds to locations outside the image space for which there are no voxels. The values for edge voxels in the smoothed image are generally reduced towards zero, an edge effect. The effect is as if a boundary of voxels with zero value were placed round the edge of the image. This scenario is zero-boundary smoothing.
**Truncated smoothing**

If the image space adequately contains the volume of interest, in our case the voxels corresponding to the brain, then the zero-boundary edge effect is not of any consequence. As can be seen from the raw images (ch.1) the brain fits just inside the image space in the X and Y directions, but is truncated in the Z direction due to the limited axial length of the tomograph.

The effect can be avoided by using *boundary-truncation* smoothing (eqn.78). Here the weights of the filter kernel are normalised at each voxel. The effect is as if the filter kernel is truncated when it reaches the edge of the image.

\[
Y_k^S = \frac{1}{\sum_{k'=1}^{K} f(x_k - x_{k'}) \sum_{k'=1}^{K} f(x_k - x_{k'})} Y_k^{'}
\]  

(78)

**B:4 Gaussian kernels**

The filter kernel used is almost universally Gaussian, by which we mean that it is the probability density function (PDF) of a \(D\)-variate normal distribution with zero mean and variance-covariance matrix \(\Sigma\) (eqn.79).

\[
f(x) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} x \Sigma^{-1} x^T \right)
\]

(79)

The filters usually used are orthogonal, with variance-covariance matrices with zero off-diagonal elements. The filter is then completely specified by the \(D\)-tuple containing the variances, and is ellipsoidal in shape, with axes parallel to the image axes. The convolution integral factorises into \(D\) one dimensional component integrals, which simplifies and speeds up computation.

**Relationship of FWHM to variance-covariance matrix**

As with image resolution, the shape of the kernel is expressed in FWHM.\(^{56}\) This is related to the variance for a Gaussian PDF as follows: A univariate Gaussian PDF with variance \(\sigma^2\) has maxima \(1/(\sigma\sqrt{2\pi})\) at \(x=0\). The FWHM \(l\) is then the width of the PDF at half this height, \(f(l/2) = 1/(2\sigma\sqrt{2\pi})\), so \(l = \sigma\sqrt{8\ln(2)}\). This is extended to \(D\)-dimensional orthogonal kernels in the obvious way. If a spherical orthogonal kernel with \(\Sigma = \sigma^2 I_D\) is used then it is common to just quote \(\sigma\sqrt{8\ln(2)}\) as the FWHM. Some authors prefer to specify FWHM in terms of voxels.

**Some common filters**

Commonly used three-dimensional filters are 10mm×10mm×12mm, and 20mm×20mm×24mm, with variance-covariance matrices of

\[
\Sigma = \begin{pmatrix} 10^2 & 0 & 0 \\ 0 & 10^2 & 0 \\ 0 & 0 & 12^2 \end{pmatrix} \frac{1}{8\ln(2)} \quad \text{and} \quad \Sigma = \begin{pmatrix} 20^2 & 0 & 0 \\ 0 & 20^2 & 0 \\ 0 & 0 & 24^2 \end{pmatrix} \frac{1}{8\ln(2)}
\]

\(^{56}\)Recall that the Full Width at Half Maximum (FWHM) is the width of the (point spread) function at half its maximum.
C: Some Results For Smoothing Convolution

The following results for smoothing convolution of random fields are useful.

C:1 Smoothing convolution: commutative for even kernels

Let \( f_1(x) \) and \( f_2(x) \) be any two even functions of \( x \in \mathbb{R}^D \). Then \( f_1 \otimes f_2 = f_2 \otimes f_1 \), that is, smoothing convolution is commutative for even functions.

**Proof:**

\[
\begin{align*}
    f_1 \otimes f_2(u) &= \int_{\mathbb{R}^D} f_2(v) f_1(u+v) dv \\
                     &= \int_{\mathbb{R}^D} f_2(w-u) f_1(w) dw \\
                     &= \int_{\mathbb{R}^D} f_2(w-u) f_1(w) dw \\
                     &= \int_{\mathbb{R}^D} f_2(u+v) f_1(-v) dv \\
                     &= \int_{\mathbb{R}^D} f_2(u+v) f_1(-v) dv \\
                     &= f_2 \otimes f_1(u)
\end{align*}
\]

So \( f_1 \otimes f_2 = f_2 \otimes f_1 \), since convolution is commutative for even kernels. The Jacobians for the changes of variables (†), are \( \det(-I_D) \), the determinant of the negative of the \( D \times D \) identity matrix, which has absolute value 1.

C:2 Double smoothing convolution: Associative for even kernels

Let \( X(x) \) be any function, \( x \in \mathbb{R}^D \), and let \( f_1(x) \) and \( f_2(x) \) be two even filter kernels. Then \( (X \otimes f_1) \otimes f_2 = X \otimes (f_1 \otimes f_2) \), that is, smoothing convolution is associative.

**Proof:**

\[
\begin{align*}
    (X \otimes f_1) \otimes f_2 &= \int_{\mathbb{R}^D} f_2(v) X \otimes f_1(x+v) dv \\
                              &= \int_{\mathbb{R}^D} \int_{\mathbb{R}^D} f_2(v) f_1(u) X(x+u+v) dudv \\
                              &= \int_{\mathbb{R}^D} f_2(v) \int_{\mathbb{R}^D} f_1(w-v) X(x+w) dw dv \\
                              &= \int_{\mathbb{R}^D} \int_{\mathbb{R}^D} f_2(v) f_1(w-v) dw X(x+w) dv \\
                              &= \int_{\mathbb{R}^D} \int_{\mathbb{R}^D} f_2(-u) f_1(w+u) dw X(x+w) dv \\
                              &= \int_{\mathbb{R}^D} \int_{\mathbb{R}^D} f_2(-u) f_1(w+u) dw X(x+w) dv
\end{align*}
\]
\[
\begin{align*}
\int \int f_2(u) f_1(w+u) \, du \, X(x+w) \, dw &= \int \int f_1 \otimes f_2(w) \, X(x+w) \, dw \tag{by even property of \( f_2 \)} \\
&= \int f_1 \otimes f_2(w) \, X(x+w) \, dw = X \otimes (f_1 \otimes f_2)
\end{align*}
\]

**C:3 Double smoothing convolution: Order unimportant**

Let \( X(x) \) be any function, \( x \in \mathbb{R}^D \), and let \( f_1(x) \) and \( f_2(x) \) be two filter kernels. Then \((X \otimes f_1) \otimes f_2 = (X \otimes f_2) \otimes f_1\), that is the order of smoothing is unimportant.

**Proof:**
For even filter kernels \( f_1 \) and \( f_2 \) this result follows as a corollary of the previous two results. However, it holds for general \( f_1 \) & \( f_2 \):

\[
\begin{align*}
(X \otimes f_1) \otimes f_2 &= \int f_2(v) \, X \otimes f_1(x+v) \, dv = \int f_2(v) \, \int f_1(u) \, X(x+u+v) \, dudv \\
&= \int f_1(u) \, \int f_2(v) \, X(x+u+v) \, dvdu = \int f_1(u) \, X \otimes f_2(x+u) \, du = (X \otimes f_2) \otimes f_1
\end{align*}
\]

**C:4 Combining Gaussian kernels: Double smoothing**

Let \( X(x) \) be any function, \( x \in \mathbb{R}^D \), and let \( f_1(x) \) and \( f_2(x) \) be two Gaussian filter kernels with variance-covariance matrices \( \Sigma_1 \) and \( \Sigma_2 \) respectively. Then \((X \otimes f_1) \otimes f_2 = (X \otimes f_2) \otimes f_1 = X \otimes (f_1 \otimes f_2) = X \otimes (f_2 \otimes f_1) = X \otimes f\) where \( f(x) \) is a Gaussian filter kernel with variance-covariance matrix \( \Sigma = \Sigma_1 + \Sigma_2 \).

**Proof:**
Since the kernels are even functions of \( x \in \mathbb{R}^D \), the associativity and commutativity properties give \((X \otimes f_1) \otimes f_2 = (X \otimes f_2) \otimes f_1 = X \otimes (f_1 \otimes f_2) = X \otimes (f_2 \otimes f_1)\). It remains to prove that \( f_1 \otimes f_2 \) has the required form. This can be done using Fourier transforms, or directly as follows:

\[
f_1 \otimes f_2(u) = \frac{1}{(2\pi)^D \sqrt{\|\Sigma_1\| \|\Sigma_2\|}} \int_{\mathbb{R}^D} \exp\left(-\frac{1}{2} \left[ v^T \Sigma_2^{-1} v + (u+v)^T \Sigma_1^{-1} (u+v) \right]\right) dv
\]

\[
= \frac{1}{(2\pi)^D \sqrt{\|\Sigma_1\| \|\Sigma_2\|}} \int_{\mathbb{R}^D} \exp\left(-\frac{1}{2} \left[ v^T (\Sigma_1^{-1} + \Sigma_2^{-1}) v + u^T \Sigma_1^{-1} u + 2v^T \Sigma_1^{-1} u \right]\right) dv
\]

Using \( u^T \Sigma_1^{-1} v = (u^T \Sigma_1^{-1} v)^T = v^T \Sigma_1^{-1} u \) (since \( \Sigma_1^{-1} \) is symmetric)
\[
\begin{align*}
\int_{\mathbb{R}^p} \exp \left( -\frac{1}{2} \left[ (v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u)^T (\Sigma_1^{-1} + \Sigma_2^{-1}) \left( v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u \right) + u^T \Sigma_1^{-1} (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u + u^T \Sigma_1^{-1} u \right] \right) dv &= \frac{1}{(2\pi)^D} \frac{1}{\sqrt{\Sigma_1 \Sigma_2}} \\
\int_{\mathbb{R}^p} \exp \left( -\frac{1}{2} \left[ (v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u)^T (\Sigma_2^{-1} (\Sigma_2 + \Sigma_1) \Sigma_1^{-1}) \left( v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u \right) \right] \right) dv \times \exp \left( -\frac{1}{2} \left[ u^T \Sigma_1^{-1} (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u \right] \right) &= \frac{1}{(2\pi)^D} \frac{1}{\sqrt{\Sigma_1 \Sigma_2}} \sqrt{\left( \Sigma_2^{-1} (\Sigma_2 + \Sigma_1) \Sigma_1^{-1} \right)^{-1}} \times \exp \left( -\frac{1}{2} \left[ u^T \Sigma_2 (\Sigma_1^{-1} + \Sigma_2^{-1}) \Sigma_1^{-1} u \right] \right) \\
\int_{\mathbb{R}^p} \exp \left( -\frac{1}{2} \left[ (v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u)^T (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_2^{-1} u \right] \right) dv &= \frac{1}{(2\pi)^D} \frac{1}{\sqrt{\Sigma_1 \Sigma_2}} \sqrt{\Sigma_1 (\Sigma_2 + \Sigma_1)^{-1} \Sigma_2} \times \exp \left( -\frac{1}{2} \left[ u^T \Sigma_2 (\Sigma_1^{-1} + \Sigma_2^{-1}) \Sigma_1^{-1} u \right] \right) \\
\int_{\mathbb{R}^p} \exp \left( -\frac{1}{2} \left[ (v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u)^T (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u \right] \right) dv &= \frac{1}{(2\pi)^D} \frac{1}{\sqrt{\Sigma_1 + \Sigma_2}} \sqrt{(\Sigma_2 + \Sigma_1)^{-1}} \times \exp \left( -\frac{1}{2} \left[ u^T (\Sigma_1 + \Sigma_2)^{-1} u \right] \right) \\
\int_{\mathbb{R}^p} \exp \left( -\frac{1}{2} \left[ (v + (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u)^T (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} u \right] \right) dv &= \frac{1}{(2\pi)^D} \frac{1}{\sqrt{\Sigma_1 + \Sigma_2}} \exp \left( -\frac{1}{2} \left[ u^T (\Sigma_1 + \Sigma_2)^{-1} u \right] \right)
\end{align*}
\]
C: Covariance function of smoothed white noise processes

Let $Z(x), x \in \mathbb{R}^D$, be the field formed by convolving a weakly stationary continuous white noise random field $X(x)$ of variance $\sigma^2$ ($\text{Var}[X(x)] = \sigma^2 \ \forall x$) with a kernel $f(x)$. Clearly $Z(x)$ is a strictly stationary continuous random field. Moreover, the covariance function is $C(h) = \sigma^2 f \otimes f(h)$

**Proof:**
This result is easily proved directly from the convolution integral (eqn.76):

$$C(h) \ = \ \text{Cov}[Z(x),Z(x+h)]$$

$$= \text{Cov}\left[\int_{\mathbb{R}^D} f(r')X(x+r')dr',\int_{\mathbb{R}^D} f(r)X(x+h+r)dr\right]$$

$$= \int_{\mathbb{R}^D} f(r)\text{Cov}\left[\int_{\mathbb{R}^D} f(r')X(x+r')dr',X(x+h+r)\right]dr$$

$$= \int_{\mathbb{R}^D} f(r)f(r + h)\sigma^2 dr$$

(since $\text{Cov}[X(x+r'),X(x+h+r)]=\sigma^2$ if $r'=h+r$ and is zero otherwise)

$$= \sigma^2 f \otimes f(h)$$

**Corollary:**

The field $Z(x)$ formed by convolving a white noise field of variance $\sigma^2$ with a Gaussian kernel $f(x)$ with variance-covariance matrix $\Sigma$, $f(x) = \exp(-x^T \Sigma^{-1} x/2) / \sqrt{(2\pi)^D |\Sigma|}$, is a strictly stationary continuous field with covariance function:

$$C(h) \ = \ \sigma^2 f \otimes f(h)$$

$$= \sigma^2 \exp(-h^T (2\Sigma)^{-1} h/2) / \sqrt{(2\pi)^D \det(2\Sigma)}$$

(a Gaussian kernel with variance-covariance $2\Sigma$, by result 4)

$$= \frac{\sigma^2}{2^D \pi^{D/2} \sqrt{|\Sigma|} \det(2\Sigma)} \exp(-h^T (\Sigma)^{-1} h/4)$$

So, for $h = 0$, $\text{Var}[Z(x)] = \frac{\sigma^2}{2^D \pi^{D/2} \sqrt{|\Sigma|}}$

**Corollary:**

A strictly stationary continuous random field with zero mean, variance $\sigma^2$, and auto-correlation function $R(h) = \exp(-h^T (\Sigma)^{-1} h/4)$, can be obtained by convolving a white noise random field of variance $\sigma^2 2^D \pi^{D/2} \sqrt{|\Sigma|}$ with a Gaussian kernel of variance-covariance matrix $\Sigma$, $f(x) = \exp(-x^T \Sigma^{-1} x/2) / \sqrt{(2\pi)^D \det(\Sigma)}$.

---

$^{57}$A random process $X(t)$ is white noise if $E[X(t)] = 0$, and if $X(t_1)$ and $X(t_2)$ are independent, for all points $t, t_1 \& t_2$ in the parameter space.
C:6 Smoothness of smoothed Gaussian white noise fields

The field obtained by convolving a continuous white noise Gaussian random field (defined on \( \mathbb{R}^D \)) with a kernel \( f(x) \), is itself a strictly stationary continuous Gaussian random field with zero mean. If the variance of the white noise process is chosen such that the resulting field has unit variance, then Adler (1981) shows that the variance-covariance matrix of partial derivatives is:

\[
\Lambda = \frac{\int_{\mathbb{R}^D} \frac{\partial f(x)}{\partial x} \cdot \frac{\partial f(x)}{\partial x^*} dx}{\int_{\mathbb{R}^D} f(x) dx}
\]

C:7 Smoothness of (Gaussian) smoothed Gaussian white noise

The strictly stationary continuous standard Gaussian (zero mean, unit variance) random field formed by convolving a white noise Gaussian random field with a Gaussian kernel, \( f(x) = \exp(-x^T \Sigma^{-1} x/2) / (2\pi)^{D/2} |\Sigma| \), has variance-covariance matrix of partial derivatives \( \Lambda = \Sigma^{-1/2} = (2\Sigma)^{-1} \). By result 5, the variance of the white noise field must be \( 2^D \pi^{D/2} / |\Sigma| \) for the smoothed field to have unit variance.

**Proof:**
This follows from the previous result by direct integration.

**Corollary:**
A strictly stationary continuous Gaussian random field with zero mean, unit variance, variance-covariance matrix of partial derivatives \( \Lambda \), and Gaussian auto-correlation function can be obtained by convolving a white noise Gaussian random field of variance \( (2\pi)^{D/2} / \sqrt{\Lambda} \) with a Gaussian kernel with variance-covariance matrix \( \Sigma = (2\Lambda)^{-1} \). This observation provides the framework for simulating Gaussian random fields.

C:8 Secondary smoothing

Consider a strictly stationary continuous standard Gaussian (zero mean, unit variance) random field \( Y(x), x \in \mathbb{R}^D \), with Gaussian auto-correlation function and variance-covariance matrix of partial derivatives \( \Lambda_Y \). Let \( f(x) \) be a Gaussian kernel with variance-covariance matrix \( \Sigma \). Let \( Z = c^{-1/2} \times Y \otimes f \), where the constant \( c \) is chosen such that \( \text{Var}(Z(x)) = 1 \ \forall x \). Then \( Z \) is also a strictly stationary standard Gaussian random field, with variance-covariance matrix of partial derivatives \( \Lambda_Z = (2\Sigma + \Lambda_Y^{-1})^{-1} \).

Furthermore, \( c = 1/\sqrt{2 \Lambda_Y \Sigma + I_D} \).

If \( \Sigma_Y = (2\Lambda_Y)^{-1} \), then \( \Lambda_Z = (2\Sigma + 2\Sigma_Y)^{-1} \), \( c = \sqrt{\Sigma_Y} / \sqrt{|\Sigma + \Sigma_Y|} \), and \( \Sigma_Z = (2\Lambda_Z)^{-1} = \Sigma + \Sigma_Y \).
Proof:

Since \( Y(x) \) can be generated by convolving a white noise Gaussian random field \( X(x) \), of variance \( (2\pi)^{D/2} / \sqrt{|\Lambda_Y|} \), with Gaussian kernel \( f_Y(x) \) with variance-covariance matrix \( \Sigma_Y = (2\Lambda_Y)^{-1} \) (corollary to result 7). Result 4 then gives that \( \sqrt{c} \times Z(x) = Y \otimes f \) is equivalent to a field obtained by convolving a white noise random field, of variance \( (2\pi)^{D/2} / \sqrt{|\Lambda_Y|} \), with a Gaussian kernel \( f_Z = f_Y \otimes f \), with variance-covariance matrix \( \Sigma + \Sigma_Y \). Then, by result 7, \( Z(x) \) has variance-covariance matrix of partial derivatives \( \Lambda_Z = (2\Sigma + 2\Sigma_Y)^{-1} = (2\Sigma + \Lambda_Y^{-1})^{-1} \). It remains to identify the constant.

Since \( Z(x) \) is of unit variance, \( \sqrt{c} \times Z(x) \) has variance \( c \), but, regarding \( \sqrt{c} \times Z(x) \) as \( X \otimes (f_Y \otimes f) \), result 5 gives its variance as:

\[
c = \frac{\sqrt{|\Sigma_Y|}}{\sqrt{|\Sigma + \Sigma_Y|}} = \frac{\sqrt{(2\Lambda_Y)^{-1}}}{\sqrt{\Sigma + (2\Lambda_Y)^{-1}}} = \frac{1}{\sqrt{2\Lambda_Y \Sigma + (2\Lambda_Y)^{-1}}} = \frac{1}{\sqrt{2\Lambda_Y \Sigma + I_D}}
\]

C:9 Effect of scaling on smoothness

Let \( Z(x), x \in \mathbb{R}^D \), be a strictly stationary continuous random field with variance-covariance matrix of partial derivatives \( \Lambda_Z \). Then the field \( Y(x) = c \times Y(x) \), for constant \( c \), has variance-covariance matrix of partial derivatives \( \Lambda_Y = c^2 \Lambda_Z \).

Proof:

Trivial. Using the definition of \( \Lambda_Z \) and the chain rule for differentiation…

\[
\Lambda = \begin{pmatrix}
\text{var}\left[ \frac{\partial Y}{\partial x_1} \right] & \text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \text{var}\left[ \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\vdots & \vdots & \ddots
\end{pmatrix}
\]

Where \( x = (x_1, x_2, \ldots) \).

\[
= \begin{pmatrix}
\text{var}\left[ \frac{\partial Y}{\partial x_1} \right] & \text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \text{var}\left[ \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\vdots & \vdots & \ddots
\end{pmatrix}
\]

\[
= \begin{pmatrix}
\text{var}\left[ \frac{\partial Y}{\partial x_1} \right] & \text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \text{var}\left[ \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\vdots & \vdots & \ddots
\end{pmatrix}
\]

\[
= \left( \frac{\partial Y}{\partial x} \right)^2 \begin{pmatrix}
\text{var}\left[ \frac{\partial Y}{\partial x_1} \right] & \text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\text{cov}\left[ \frac{\partial Y}{\partial x_1}, \frac{\partial Y}{\partial x_2} \right] & \text{var}\left[ \frac{\partial Y}{\partial x_2} \right] & \cdots \\
\vdots & \vdots & \ddots
\end{pmatrix} = c^2 \Lambda
\]
D: Expected Euler Characteristics

D:1 The $\chi^2$-field

The expected Euler characteristic $\chi(A_u(U,\Psi))$ of the excursion set $A_u(U,\Psi)$ of a homogeneous (strictly stationary) Chi-squared random field $U(x)$ with $n$ degrees of freedom, defined on $x \in \Psi$, a compact, convex subset of $\mathbb{R}^D$ (with boundary of zero Lebesgue measure), for a threshold $u$ is, for $D \geq 2$ and under mild conditions on the component fields:

$$E[\chi(A_u(U,\Psi))] = \frac{\lambda(\Psi)|A|^{\frac{1}{2}} u^{\frac{D}{2}} \exp\left(-\frac{u}{4}\right)}{(2\pi)^{\frac{D}{2}} 2^{\frac{D+1}{2}} \Gamma\left(\frac{D}{2}\right)} P_{D,n}(u)$$

where $P_{D,n}(u)$ is a polynomial of degree $D-1$ in $u$ with integer coefficients:

$$P_{D,n}(u) = \sum_{j=0}^{n+1} \sum_{k=0}^{D-1-2j} \binom{n-1}{D-1-2j-k} (-1)^{D-1+j+k} (D-1)! \frac{2^j j! k!}{(D-1-2j-k)!} u^{j+k}$$

Terms with factorials of negatives in the denominator are taken as zero. (Worsley, 1994, Theorem 3.5)

D:2 The $F$-field

For $F(x)$ an $F$-field with $n,m$ degrees of freedom, the expected Euler characteristic of the excursion set (over $\Psi$, a compact, convex subset of $\mathbb{R}^D$, with boundary of zero Lebesgue measure) above a threshold $u$ is, for $m+n > D \geq 2$:

$$E[\chi(A_u(F,\Psi))] = \frac{\lambda(\Psi)|A|^{\frac{1}{2}} \Gamma\left(\frac{1}{2} (m+n-D)\right) \left(\frac{nu}{m}\right)^{\frac{m-D}{2}} \Gamma\left(\frac{m+n-D}{2}\right) \Gamma\left(\frac{n}{2}\right) \Gamma\left(\frac{m}{2}\right)}{(2\pi)^{\frac{D}{2}} 2^{\frac{D}{2}} \Gamma\left(\frac{D}{2}\right) \Gamma\left(\frac{n}{2}\right)} K_{D,m,n}(u)$$

where $K_{D,m,n}(u)$ is a polynomial of degree $D-1$ in $nu/m$ with integer coefficients:

$$K_{D,m,n}(u) = (-1)^{D-1} (D-1)! \sum_{j=0}^{n+1} \frac{\Gamma\left(\frac{1}{2} (m+n-D)+j\right)}{\Gamma\left(\frac{1}{2} (m+n-D)\right) j!} \times \sum_{k=0}^{D-1-2j} \binom{n-1}{D-1-2j-k} (-1)^{j+k} \left(\frac{nu}{m}\right)^{j+k}$$

(Worsley, 1994, Theorem 4.6)
D:3 The $t$-field

For $T(x)$ a $t$-field with $n$ degrees of freedom, the expected Euler characteristic of the excursion set (over $\Psi$, a compact, convex subset of $\mathbb{R}^D$, with boundary of zero Lebesgue measure) above a threshold $u$ is, for $n \geq D \geq 2$:

$$E[\chi(A_u(T, \Psi))] = \frac{\lambda(\Psi) |\lambda|^{1/2}}{(2\pi)^{D/2}} \left(1 + \frac{u^2}{n}\right)^{-\frac{n-1}{2}} Q_{D,n}(u)$$

where $Q_{D,n}(u)$ is a polynomial of degree $D - 1$ in $u$:

$$Q_{D,n}(u) = \sum_{j=0}^{D-1} \frac{(-1)^j (D-1)!}{2^j j!(D-1-2j)!} \frac{\Gamma\left(\frac{1}{2} (n+1)\right)}{\Gamma\left(\frac{1}{2} (n+2-D+2j)\right)(\frac{1}{2} n)^{-\frac{1}{2}}} u^{D-1-2j}$$

(Worsley, 1994, Theorem 5.3)
E: “Transform” Functions

Suppose \( t \) is drawn from a distribution with Cumulative Distribution Function \( F(t) \), then an equivalent standard Gaussian variate \( z \) is one with equal extremum probability: \[ \Phi(z) = F(t) \iff z = \Phi^{-1}(F(t)) \]

(The distribution function method for functions of random variables.) Since the normal distribution is continuous, \( \Phi(z) \) is strictly monotonic increasing. Thus \( \Phi^{-1} \) exists and a unique \( z \) is specified. \( \Phi^{-1}(F(t)) \) is thus a function “transforming” a random variable from one distribution to a standard Gaussian distribution, and has become known (in PET) as a transform function.

Transform function for Students’ \( t \)-distribution

The computing environments used by most PET centres to analyse images do not have built in statistical distribution functions (PDFs, CDFs, inverse CDFs), so they must be explicitly coded. Since the evaluation of these functions is becoming a lost art in these days of comprehensive tables and sophisticated statistics packages, we review the computations for Students \( t \)-distribution.

For the \( t \)-distribution with \( df \) degrees of freedom, the Cumulative Distribution Function \( F_T(t) \) can be expressed in terms of the incomplete Beta function, \( I_x(a,b) \), as follows:

\[
F_T(t) = \frac{1}{\sqrt{df} \beta\left(\frac{df}{2}, \frac{1}{2}\right)} \int_0^t \left(1 + \frac{u^2}{df}\right)^{-\frac{df+1}{2}} du = \begin{cases} 
1 - \frac{1}{2} \times I\left(\frac{df}{2}, \frac{1}{2}\right) & t \geq 0 \\
\frac{1}{2} \times I\left(\frac{df}{2}, \frac{1}{2}\right) & t \leq 0
\end{cases}
\]

Here \( \beta(a,b) \) is Beta function:

\[
\beta(a,b) = \frac{\Gamma(a) \Gamma(b)}{\Gamma(a+b)} = \int_0^1 u^{a-1} (1-u)^{b-1} du \quad a, b > 0
\]

and the incomplete Beta function is:

\[
I_x(a,b) = \frac{\beta_x(a,b)}{\beta(a,b)} = \frac{1}{\beta(a,b)} \int_0^x u^{a-1} (1-u)^{b-1} du \quad a, b > 0
\]

As is well known, the CDF of the normal distribution is related to the error function:

\[
\Phi(x) = \frac{1}{2} + \frac{1}{2} \text{erf}\left(\frac{x}{\sqrt{2}}\right) \quad \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du
\]

\[
\Phi^{-1}(p) = \sqrt{2} \text{erf}^{-1}(2p - 1) \quad 0 \leq p \leq 1
\]

The incomplete Beta function and the inverse of the error function are supplied in many engineering and imaging packages. Alternatively, various published solutions for their approximation exist. Thus the “transform” function for Students \( t \)-distribution can be painlessly coded. See “Numerical Recipes” (Press et. al.) for algorithms and relationships for other distributions.
**F: Ordering Theorem**

**Theorem**

Consider two sets of real numbers: \( X = \{x_i\}_{i=1}^n \) and \( Y = \{y_j\}_{j=1}^n \), such that \( x_i \leq y_i \ \forall \ i = 1, \ldots, n \). Order \( X \) and \( Y \) from largest to smallest, with ties broken arbitrarily, giving ordered sets \( X = \{x_{(i)}\}_{i=1}^n \) and \( Y = \{y_{(j)}\}_{j=1}^n \), where \( x_{(i)} \geq x_{(j)} \) and \( y_{(i)} \geq y_{(j)} \ \forall \ 1 \leq i \leq j \leq n \).

Then, \( x_{(i)} \leq y_{(i)} \ \forall \ i = 1, \ldots, n \).

**Proof:**

Suppose \( x_{(k)} = x_{i_k}, \ y_{(k)} = y_{j_k} \)

and \( S_{X,k} = \{i_l : l = 1, \ldots, k\} \), the \( k \) largest \( x_i \), and \( S_{Y,k} = \{j_l : l = 1, \ldots, k\} \), the \( k \) largest \( y_j \).

\( k = 1 \)

\[ y_{(1)} = y_{j_1} \geq y_{i_1} \quad \text{by definition of maximum} \]
\[ \geq x_{i_1} \quad \text{by hypothesis} \]
\[ = x_{(1)} \quad \text{by definition of } x_{i_1} \]

\( k > 1 \)

If \( i_k \notin S_{Y,k-1} \) then \( y_{(k)} \geq y_{j_k} \quad (i_k \notin S_{Y,k-1} \Rightarrow y_{j_k} \text{ not in } k-1 \text{ largest } y) \)
\[ \geq x_{i_k} \quad \text{by hypothesis} \]
\[ = x_{(k)} \quad \text{by definition of } x_{i_k} \]

otherwise, \( i_k \in S_{Y,k-1} \), and \( \exists \ i' \) such that \( i' \in S_{X,k-1} \) but \( i' \notin S_{Y,k-1} \)

then \( y_{(k)} \geq y_{j'} \quad \text{because } i' \notin S_{Y,k-1} \)
\[ \geq x_{i'} \quad \text{by hypothesis} \]
\[ \geq x_{(k)} \quad \text{because } i' \in S_{X,k-1} \]

\( \dagger \) \( S_{X,k-1} \) has \( k-1 \) members, but \( i_k \) is not one of them, by definition. However, \( S_{Y,k-1} \) has \( k-1 \) members, one of which is \( i_k \). Therefore \( S_{X,k-1} \) must contain an element not contained in \( S_{Y,k-1} \).
G: Smoothness of $t$-Fields

Let $X_1(x), \ldots, X_n(x) \in \Psi \subset \mathbb{R}^D$ be independent, identically distributed, strictly stationary Gaussian random fields with zero mean and variance $\sigma^2$. Suppose that the variance-covariance matrix of partial derivatives of the field is $\Lambda$.

Consider the $t$-field $T(x)$ with $n-1$ degrees of freedom formed as the one-sample $t$-statistic of $\{X_1(x), \ldots, X_n(x)\}$ at each point $x \in \Psi$: \[ T(x) = \frac{M(x)}{S(x)/\sqrt{n}} \]

where \[ M(x) = \frac{1}{n} \sum_{i=1}^{n} X_i(x) \]

and \[ S(x)^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i(x) - M(x))^2 \]

Let $\Lambda_T$ be the variance-covariance matrix of partial derivatives of the $t$-field. This can be related to $\Lambda$ using a simplification of the argument used in the appendix of Worsley et al. (1992). (With thanks to Dr. Worsley for pointing this out.) The argument is as follows:

Since $X_1(x), \ldots, X_n(x)$ and their partial derivatives are all independent Gaussian random variables with zero expectations (Adler, 1981, p.31), conditioning on $X_1(x), \ldots, X_n(x)$, we have

\[ \Lambda_T = \text{Var} \left( \frac{\partial T}{\partial x} \right) = \text{Var} \left( \sum_{i=1}^{n} \left( \frac{\partial T}{\partial X_i} \right) \left( \frac{\partial X_i}{\partial x} \right) \right) \]

\[ = \mathbb{E} \left[ \sum_{i=1}^{n} \left( \frac{\partial T}{\partial X_i} \right)^2 \text{Var} \left( \frac{\partial X_i}{\partial x} \right) \right] \]

\[ = \mathbb{E} \left[ \sum_{i=1}^{n} \left( \frac{\partial T}{\partial X_i} \right)^2 \right] \Lambda \sigma^2 \]

From the definition of $T(x)$ we have \[ \frac{\partial T}{\partial X_i} = \frac{\partial}{\partial X_i} \frac{M}{S/\sqrt{n}} = \frac{1}{n^{1/2}} \frac{n^{1/2} M(X_i - M)}{(n-1) S^3} \]

and so

\[ \sum_{i=1}^{n} \left( \frac{\partial T}{\partial X_i} \right) = \frac{1}{S^2} + \frac{nm^2}{(n-1) S^4} \]
Hence

\[ \Lambda_T = \mathbb{E} \left[ \frac{1}{S^2} + \frac{nM^2}{(n-1)S^4} \right] \Lambda \sigma^2 \]

\[ = \lambda_n \Lambda \quad \text{say.} \]

This can be further simplified using the fact that \( U = \frac{nM^2 + (n-1)S^2}{\sigma^2} \) has a \( \chi^2 \) distribution with \( n \) degrees of freedom, independent of \( T \), and that \( \mathbb{E}[1/U^2] = 1/(n-2) \), giving

\[ \lambda_n = \mathbb{E} \left[ \frac{(T^2 + n - 1)^2}{(n-1)(n-2)} \right] \]

Integrating over the density of \( T \) gives:

\[ \lambda_n = \int_{-\infty}^{+\infty} \left( \frac{T^2 + n - 1}{(n-1)(n-2)} \right) f_T(t) \, dt = 2 \int_{0}^{+\infty} \left( \frac{(t^2 + n - 1)^2}{(n-1)(n-2)} \right) f_T(t) \, dt \]

where \( f_T(\bullet) \) is the PDF of a Student’s \( t \)-distribution with \( n-1 \) degrees of freedom. The integral is finite only for \( n \geq 4 \) (\( df \geq 3 \)).

Values of \( \lambda_n \) for \( n = 4, \ldots, 199 \) are given to 4dp in the following table, computed using an adaptive recursive Newton-Cotes eight panel rule. Note that \( \lambda_n \) tends to 1 from above as \( n \) tends to infinity.

<table>
<thead>
<tr>
<th>( \lambda_n )</th>
<th>+0</th>
<th>+1</th>
<th>+2</th>
<th>+3</th>
<th>+4</th>
<th>+5</th>
<th>+6</th>
<th>+7</th>
<th>+8</th>
<th>+9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1.5243</td>
<td>1.4467</td>
<td>1.3891</td>
<td>1.3446</td>
<td>1.3092</td>
<td>1.2804</td>
<td>1.2565</td>
<td>1.2363</td>
<td>1.2191</td>
<td>1.2042</td>
</tr>
<tr>
<td>20</td>
<td>1.1912</td>
<td>1.1798</td>
<td>1.1696</td>
<td>1.1605</td>
<td>1.1524</td>
<td>1.1450</td>
<td>1.1383</td>
<td>1.1323</td>
<td>1.1267</td>
<td>1.1215</td>
</tr>
<tr>
<td>30</td>
<td>1.1168</td>
<td>1.1124</td>
<td>1.1083</td>
<td>1.1045</td>
<td>1.1010</td>
<td>1.0977</td>
<td>1.0946</td>
<td>1.0918</td>
<td>1.0890</td>
<td>1.0865</td>
</tr>
<tr>
<td>40</td>
<td>1.0840</td>
<td>1.0818</td>
<td>1.0796</td>
<td>1.0775</td>
<td>1.0756</td>
<td>1.0737</td>
<td>1.0720</td>
<td>1.0703</td>
<td>1.0687</td>
<td>1.0671</td>
</tr>
<tr>
<td>50</td>
<td>1.0657</td>
<td>1.0643</td>
<td>1.0629</td>
<td>1.0616</td>
<td>1.0604</td>
<td>1.0592</td>
<td>1.0580</td>
<td>1.0569</td>
<td>1.0559</td>
<td>1.0549</td>
</tr>
<tr>
<td>60</td>
<td>1.0539</td>
<td>1.0529</td>
<td>1.0520</td>
<td>1.0511</td>
<td>1.0503</td>
<td>1.0494</td>
<td>1.0486</td>
<td>1.0479</td>
<td>1.0471</td>
<td>1.0464</td>
</tr>
<tr>
<td>70</td>
<td>1.0457</td>
<td>1.0450</td>
<td>1.0443</td>
<td>1.0437</td>
<td>1.0431</td>
<td>1.0424</td>
<td>1.0419</td>
<td>1.0413</td>
<td>1.0407</td>
<td>1.0402</td>
</tr>
<tr>
<td>80</td>
<td>1.0396</td>
<td>1.0391</td>
<td>1.0386</td>
<td>1.0381</td>
<td>1.0376</td>
<td>1.0372</td>
<td>1.0367</td>
<td>1.0363</td>
<td>1.0358</td>
<td>1.0354</td>
</tr>
<tr>
<td>90</td>
<td>1.0350</td>
<td>1.0346</td>
<td>1.0342</td>
<td>1.0338</td>
<td>1.0335</td>
<td>1.0331</td>
<td>1.0327</td>
<td>1.0324</td>
<td>1.0320</td>
<td>1.0317</td>
</tr>
<tr>
<td>100</td>
<td>1.0314</td>
<td>1.0310</td>
<td>1.0307</td>
<td>1.0304</td>
<td>1.0301</td>
<td>1.0298</td>
<td>1.0295</td>
<td>1.0292</td>
<td>1.0289</td>
<td>1.0287</td>
</tr>
<tr>
<td>110</td>
<td>1.0284</td>
<td>1.0281</td>
<td>1.0279</td>
<td>1.0276</td>
<td>1.0274</td>
<td>1.0271</td>
<td>1.0269</td>
<td>1.0266</td>
<td>1.0264</td>
<td>1.0262</td>
</tr>
<tr>
<td>120</td>
<td>1.0259</td>
<td>1.0257</td>
<td>1.0255</td>
<td>1.0253</td>
<td>1.0251</td>
<td>1.0249</td>
<td>1.0247</td>
<td>1.0245</td>
<td>1.0243</td>
<td>1.0241</td>
</tr>
<tr>
<td>130</td>
<td>1.0239</td>
<td>1.0237</td>
<td>1.0235</td>
<td>1.0233</td>
<td>1.0231</td>
<td>1.0230</td>
<td>1.0228</td>
<td>1.0226</td>
<td>1.0224</td>
<td>1.0223</td>
</tr>
<tr>
<td>140</td>
<td>1.0221</td>
<td>1.0219</td>
<td>1.0218</td>
<td>1.0216</td>
<td>1.0215</td>
<td>1.0213</td>
<td>1.0212</td>
<td>1.0210</td>
<td>1.0209</td>
<td>1.0207</td>
</tr>
<tr>
<td>150</td>
<td>1.0206</td>
<td>1.0205</td>
<td>1.0203</td>
<td>1.0202</td>
<td>1.0200</td>
<td>1.0199</td>
<td>1.0198</td>
<td>1.0196</td>
<td>1.0195</td>
<td>1.0194</td>
</tr>
<tr>
<td>160</td>
<td>1.0193</td>
<td>1.0191</td>
<td>1.0190</td>
<td>1.0189</td>
<td>1.0188</td>
<td>1.0187</td>
<td>1.0186</td>
<td>1.0184</td>
<td>1.0183</td>
<td>1.0182</td>
</tr>
<tr>
<td>170</td>
<td>1.0181</td>
<td>1.0180</td>
<td>1.0179</td>
<td>1.0178</td>
<td>1.0177</td>
<td>1.0176</td>
<td>1.0175</td>
<td>1.0174</td>
<td>1.0173</td>
<td>1.0172</td>
</tr>
<tr>
<td>180</td>
<td>1.0171</td>
<td>1.0170</td>
<td>1.0169</td>
<td>1.0168</td>
<td>1.0167</td>
<td>1.0166</td>
<td>1.0165</td>
<td>1.0164</td>
<td>1.0163</td>
<td>1.0162</td>
</tr>
<tr>
<td>190</td>
<td>1.0162</td>
<td>1.0161</td>
<td>1.0160</td>
<td>1.0159</td>
<td>1.0159</td>
<td>1.0158</td>
<td>1.0157</td>
<td>1.0156</td>
<td>1.0156</td>
<td>1.0154</td>
</tr>
</tbody>
</table>
H: Poline’s Bivariate Approach

Poline & Mazoyer (1994a) address the problem of intense focal activations being missed by suprathreshold cluster size tests with low thresholds, by including the mean voxel value of the suprathreshold cluster into the testing procedure.

Methodology

The bivariate parameter space \( P = \mathbb{Z}^+ \times \mathbb{R}^+ = \{0, 1, \ldots \} \times [0, \infty) \) for the size (in voxels) and magnitude of a suprathreshold cluster is partitioned into rejection and acceptance regions by an “iso-cumulative” curve. Let \( C(s, m) \) be the number of clusters with size \( > s \) and mean voxel value \( > m \) in an single statistic image, with expected value \( E[C(s, m)] \). Let \( P_{\text{inf}}(s, m) = \{(s', m') \in P : E[C(s', m')] \leq E[C(s, m)]\} \subseteq P \) for \( (s, m) \in P \). The boundary of this region \( \{(s', m') \in P : E[C(s', m')] = E[C(s, m)]\} \) is an iso-cumulative curve, so called since for each point \( (s', m') \) on the curve, the expected number of clusters with size \( > s' \) and mean voxel value \( > m' \) is constant. Let \( I(s, m) \) be the number of suprathreshold clusters in an image with attributes \( (s', m') \in P_{\text{inf}}(s, m) \). The rejection region for a level \( \alpha \) test is then \( P_{\text{inf}}(s_{\alpha}, m_{\alpha}) \), where \( (s_{\alpha}, m_{\alpha}) \) are chosen such that \( \Pr(I(s_{\alpha}, m_{\alpha}) \geq 1) = \alpha \) under \( H_W \). The rejection region is unique, although \( (s_{\alpha}, m_{\alpha}) \) are not.

Considering the testing of a suprathreshold cluster with attributes \( (s, m) \), if \( \Pr(I(s, m) \geq 1) \leq \alpha \) then \( P_{\text{inf}}(s, m) \subseteq P_{\text{inf}}(s_{\alpha}, m_{\alpha}) \), since the iso-cumulative curves do not cross. Since \( (s, m) \in P_{\text{inf}}(s, m) \), \( (s, m) \) is in the rejection region and the omnibus null hypothesis for the suprathreshold cluster of voxels is rejected.

Since direct estimation of the rejection region from simulated statistic images is difficult, Poline & Mazoyer (1994) assumed a Poisson distribution for \( I(s, m) \), suggested by the law of rare events. Then, \( \Pr(I(s, m) \geq 1) = 1 - e^{-E[I(s, m)]} \), and only \( E[I(s, m)] \) needs to be estimated to apply the test.

It remains to estimate \( E[I(s, m)] \). For a given image, an empirical iso-cumulative curve can be computed as the boundary of \( \hat{P}_{\text{inf}}(s, m) = \{(s', m') \in P : C(s', m') \leq C(s, m)\} \), a step function, passing through \( (s, m) \). This gives an estimate, \( \hat{I}(s, m) \), of \( I(s, m) \) as the number of objects in image with attributes \( (s', m') \in \hat{P}_{\text{inf}}(s, m) \). Computing the mean of \( \hat{I}(s, m) \) over many simulated images gives an estimate of \( E[I(s, m)] \).

Comments

The description given above differs slightly to that of Poline & Mazoyer (1994a), which was not as rigorous in its definition of the iso-cumulative curves. A summary of the approach appears in Poline & Mazoyer (1994b).

A simpler approach would be to approximate the iso-cumulative curves with some function \( f \), as \( f(s, m) = c \). The function \( f \) then constitutes a statistic describing a suprathreshold portion of the statistic image. The null distribution of \( f_{\text{max}} \), the largest \( f \)-value for clusters in a single image is then easily simulated, and the appropriate quantile for a level \( \alpha \) test based on the \( f \)-values of clusters estimated. This avoids the necessity of the Poisson assumption and the lengthy computation of \( \hat{I}(s, m) \) for each object from the simulated data.
Hierarchical decomposition

In addition to the bivariate approach for suprathreshold clusters, Poline & Mazoyer (1994a) propose a hierarchical decomposition of the image into objects whose size and mean amplitude are analysed, thus avoiding having to choose a threshold for cluster identification. Essentially the local maxima are iteratively “cut off” to form the objects. Fig. 93 illustrates the objects after hierarchical decomposition of a simple 1D image. (For rigorous definitions and further explanation the reader is referred to Poline & Mazoyer (1994a) and the references there.) This represents an interesting direction. However, the hierarchical nature of the decomposition leads to objects whose attributes are not independent. The effect of this is probably negligible.

Figure 93
Hierarchical decomposition of a continuous one-dimensional “image”.
The means and sizes for “objects” C & D are shown.
References


Filliben JJ (1975) “The Probability Plot Correlation Coefficient Test for Normality” Technometrics 17:111


References


Functional Connectivity: The Principal Component Analysis of Large (PET) Da


Herman GT (1980) Image Reconstruction from Projections New York: Academic


Sokolov L (1979) “Relation between Physiological Function and Energy Metabolism in the Central Nervous System” *Journal of Neurochemistry* **29**:13–26


Tennent RM (1971) *Science Data Book* Oliver and Boyd


Worsley KJ (1994c) “Quadratic tests for local changes in random fields with applications to medical images” Submitted to Journal of the American Mathematical Association

