

APPENDIX TO PART I

In this Appendix, designated as **A1** (appendices **A2** and **A3** are for Parts II and III, respectively), we shall again refer to equations in the text by section and equation number, so that (2.4.3) refers to expression (3) in section 2.4 of Part I. Also, references to previous expressions in this Appendix (A1), will be written the same way, so that (A1.1.3) refers to expression (3) of section 1 in Appendix A1.

A1.1. Poisson Approximation of the Binomial

This standard result appears in many elementary probability texts [such as Larsen and Marx (2001, p.247)]. Here one starts with the fundamental limit identity

$$(A1.1.1) \quad \lim_{m \rightarrow \infty} \left(1 + \frac{x}{n}\right)^n = e^x$$

that defines the *exponential function*. Given this relation, observe that since

$$(A1.1.2) \quad \frac{n!}{k!(n-k)!} = \frac{n(n-1)\cdots(n-k+1)(n-k)!}{k!(n-k)!} = \frac{n(n-1)\cdots(n-k+1)}{k!}$$

it follows that expression (2.2.3) can be written as

$$(A1.1.3) \quad \begin{aligned} & \frac{n!}{k!(n-k)!} \left(\frac{a(C)}{a(R)}\right)^k \left(1 - \frac{a(C)}{a(R)}\right)^{n-k} \\ &= \left(\frac{n^k}{n^k}\right) \frac{n(n-1)\cdots(n-k+1)}{k!} \left(\frac{a(C)}{a(R)}\right)^k \left(1 - \frac{a(C)}{a(R)}\right)^{n-k} \\ &= \left(\frac{n}{n} \cdot \frac{n-1}{n} \cdots \frac{n-k+1}{n}\right) \frac{\{[n/a(R)]a(C)\}^k}{k!} \left(1 - \frac{a(C)}{a(R)}\right)^n \left(1 - \frac{a(C)}{a(R)}\right)^{-k} \end{aligned}$$

But if we now evaluate expression (A1.1.3) at the sequence in (2.3.2) and recall that $n_m/a(R_m) \rightarrow \lambda > 0$, then in the limit we can replace $n_m/a(R_m)$ by λ in the second factor. Moreover, since $(n_m - h)/n_m \rightarrow 1$ for all $h = 0, 1, \dots, k-1$, it also follows that the first factor in (A1.1.3) goes to one. In addition, the last factor also goes to one since $a(R_m) \rightarrow \infty \Rightarrow a(C)/a(R_m) \rightarrow 0$. Hence by taking limits we see that

$$(A1.1.4) \quad \lim_{m \rightarrow \infty} \frac{n_m!}{k!(n_m-k)!} \left(\frac{a(C)}{a(R_m)}\right)^k \left(1 - \frac{a(C)}{a(R_m)}\right)^{n_m-k}$$

$$\begin{aligned}
&= (1) \frac{\lambda^k}{k!} \cdot \left\{ \lim_{m \rightarrow \infty} \left(1 - \frac{a(C)}{a(R_m)} \right)^{n_m} \right\} (1) \\
&= \frac{\lambda^k}{k!} \cdot \left\{ \lim_{m \rightarrow \infty} \left(1 - \frac{a(C)[n_m / a(R_m)]}{n_m} \right)^{n_m} \right\} \\
&= \frac{\lambda^k}{k!} \cdot \left\{ \lim_{m \rightarrow \infty} \left(1 - \frac{\lambda \cdot a(C)}{n_m} \right)^{n_m} \right\} \\
&= \frac{\lambda^k}{k!} e^{-\lambda a(C)}
\end{aligned}$$

A1.2. Distributional Properties of Nearest-Neighbor Distances under CSR

Given that the nn-distance, D , for a randomly selected point has cdf

$$(A1.2.1) \quad F_D(d) = 1 - \Pr(D > d) = 1 - e^{-\lambda \pi d^2}$$

By differentiating (A1.2.1) we obtain the probability density f_D of D as

$$(A1.2.2) \quad f_D(d) = F'_D(d) = 2\lambda \pi d e^{-\lambda \pi d^2}$$

This distribution is thus seen to be an instance of the *Rayleigh distribution* (as for example in Johnson and Kotz, 1970, p.197). This distribution is closely related to the normal distribution, which can be used to calculate its moments. To do so, recall first from the symmetry of the density for the normal distribution, $N(0, \sigma^2)$, that

$$(A1.2.3) \quad \frac{1}{\sqrt{2\pi\sigma}} \int_0^\infty e^{-x^2/2\sigma^2} dx = \frac{1}{2} \Rightarrow \int_0^\infty e^{-x^2/2\sigma^2} dx = \frac{1}{2} \sqrt{2\pi\sigma}$$

Hence by setting $\sigma^2 = 1/(2\lambda\pi)$ so that $\lambda\pi = 1/(2\sigma^2)$, we obtain the identity

$$(A1.2.4) \quad \int_0^\infty e^{-\lambda\pi x^2} dx = \frac{1}{2} \sqrt{\frac{2\pi}{2\lambda\pi}} = \frac{1}{2\sqrt{\lambda}}$$

Next observe from the symmetry of the second-moment for $N(0, \sigma^2)$ that

$$(A1.2.5) \quad \frac{1}{\sqrt{2\pi\sigma}} \int_0^\infty x^2 e^{-x^2/2\sigma^2} dx = \frac{\sigma^2}{2} \Rightarrow \int_0^\infty x^2 e^{-x^2/2\sigma^2} dx = \frac{\sigma^2}{2} \sqrt{2\pi\sigma}$$

so that again by setting $\sigma^2 = 1/(2\lambda\pi)$ we obtain

$$(A1.2.6) \quad \int_0^\infty x^2 e^{-\lambda\pi x^2} dx = \frac{1}{4\lambda\pi} \sqrt{\frac{2\pi}{2\lambda\pi}} = \frac{1}{4\lambda\pi\sqrt{\pi}}$$

So to obtain the *mean*, $E(D)$, of D observe from (A1.2.2) and (A1.2.6) that

$$(A1.2.7) \quad \begin{aligned} E(D) &= \int_0^\infty x f_D(x) dx = \int_0^\infty x (2\lambda\pi x e^{-\lambda\pi x^2}) dx = 2\lambda\pi \int_0^\infty x^2 e^{-\lambda\pi x^2} dx \\ &= \frac{2\lambda\pi}{4\lambda\pi\sqrt{\pi}} = \frac{1}{2\sqrt{\pi}} \end{aligned}$$

To obtain the *variance*, $\text{var}(D)$, of D we first calculate the second moment, $E(D^2)$. To do so, observe first from the integration-by-parts identity (as for example in Bartle, 1975, Section 22) that for any differentiable functions, $f(x)$ and $g(x)$ on $[0, \infty)$,

$$(A1.2.8) \quad \int_0^\infty f(x)g'(x)dx + \int_0^\infty f'(x)g(x)dx = -f(0)g(0) + \lim_{x \rightarrow \infty} f(x)g(x)$$

whenever these integrals and limits exist. Hence letting $f(x) = x^2$ and $g(x) = e^{-\lambda\pi x^2}$, it follows that

$$(A1.2.9) \quad \int_0^\infty x^2 (-2\lambda\pi x e^{-\lambda\pi x^2}) dx + \int_0^\infty (2x)(e^{-\lambda\pi x^2}) dx = -(0) + \lim_{x \rightarrow \infty} x^2 e^{-\lambda\pi x^2} = 0$$

But by (A1.2.2) we have,

$$(A1.2.10) \quad \int_0^\infty f_D(x) dx = 1 \Rightarrow 2\lambda\pi \int_0^\infty x e^{-\lambda\pi x^2} dx = 1 \Rightarrow \int_0^\infty 2x e^{-\lambda\pi x^2} dx = \frac{1}{\lambda\pi}$$

which together with (A1.2.9) now shows that

$$(A1.2.11) \quad E(D^2) = \int_0^\infty x^2 f_D(x) dx = \int_0^\infty x^2 (2\lambda\pi x e^{-\lambda\pi x^2}) dx = \int_0^\infty 2x e^{-\lambda\pi x^2} dx = \frac{1}{\lambda\pi}$$

Finally, by combining (A1.2.7) and (A1.2.11) we obtain

$$(A1.2.12) \quad \text{var}(D) = E(D^2) - [E(D)]^2 = \frac{1}{\lambda\pi} - \left(\frac{1}{2\sqrt{\pi}} \right)^2 = \frac{1}{\lambda\pi} - \left(\frac{1}{4\lambda} \right) = \frac{4 - \pi}{4\lambda\pi}$$

A1.3. Distribution of Skellam's Statistic under CSR

Given these distributional properties of D , we next derive the distribution of *Skellam's statistic* in (3.2.6). To do so, we first observe from expression (A1.2.1) above that since the cdf of the exponential distribution with mean $1/\theta$ is given by $F(x;\theta) = 1 - e^{-\theta x}$, it follows at once that D^2 is *exponentially distributed* with mean $1/\lambda\pi$. But since sums of m independent and identically distributed exponentials with means $1/\theta$ is well known to be *Gamma distributed*, $\Gamma(m, \theta)$, (as for example in Johnson and Kotz, 1970, Chapter 17), it then follows that under CSR, the distribution of m independent nn-distance samples (D_1, \dots, D_n) , is given by,

$$(A1.3.1) \quad W_m = \sum_{i=1}^m D_i^2 \sim \Gamma(m, \lambda\pi)$$

For practical testing purposes, this is usually rescaled. Given that the gamma density for W_m has the explicit form,

$$(A1.3.2) \quad f_{W_m}(w) = \frac{(\lambda\pi)^m w^{m-1}}{(m-1)!} e^{-\lambda\pi w}$$

the change of variables

$$(A1.3.3) \quad S_m = 2\lambda\pi W_m = 2\lambda\pi \sum_{i=1}^m D_i^2$$

yields a new density

$$(A1.3.4) \quad g_{S_m}(s) = f_{W_m}(w(s)) |w'(s)| = f_{W_m}(s/2\lambda\pi) |1/2\lambda\pi|$$

$$= \frac{(\lambda\pi)^m (s/2\lambda\pi)^{m-1}}{(m-1)!} e^{-\lambda\pi(s/2\lambda\pi)} \left(\frac{1}{2\lambda\pi} \right) = \frac{2^{-m} s^{m-1}}{(m-1)!} e^{-(s/2)}$$

which is precisely the *chi-square distribution* with $2m$ degrees of freedom. Hence

$$(A1.3.5) \quad S_m = 2\lambda\pi \sum_{i=1}^m D_i^2 \sim \chi_{2m}^2$$

A1.4. Effects of Positively Dependent Nearest-Neighbor Samples

In this section it is shown that positive dependencies among nearest neighbors have the effect of *increasing* the variance of the test statistic, Z_n , thus making outlier values more

likely than they would otherwise be. To show this, suppose first that the sample nn-distance values (D_1, \dots, D_n) are identically distributed with mean, $\mu = E(D_i)$, and variance, $\sigma^2 = \text{var}(D_i) = E[(D_i - \mu)^2]$. Then as a generalization of expression (3.2.11) in the text, we have

$$\begin{aligned}
 \text{(A1.4.1)} \quad \text{var}(\bar{D}_n) &= E[(\bar{D}_n - \mu)^2] \\
 &= E\left[\left(\frac{1}{n} \sum_{i=1}^n D_i - \mu\right)^2\right] = E\left[\left(\frac{1}{n} \sum_{i=1}^n D_i - \frac{1}{n} \sum_{i=1}^n \mu\right)^2\right] \\
 &= E\left[\left(\frac{1}{n} \sum_{i=1}^n (D_i - \mu)\right)^2\right] = E\left[\frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n (D_i - \mu)(D_j - \mu)\right] \\
 &= \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n E[(D_i - \mu)(D_j - \mu)] \\
 &= \frac{1}{n^2} \sum_{i=1}^n E[(D_i - \mu)^2] + \frac{1}{n^2} \sum_{i=1}^n \sum_{j \neq i} E[(D_i - \mu)(D_j - \mu)] \\
 &= \frac{1}{n^2} \sum_{i=1}^n \text{var}(D_i) + \frac{1}{n^2} \sum_{i=1}^n \sum_{j \neq i} \text{cov}(D_i, D_j) \\
 &= \frac{1}{n^2} \sum_{i=1}^n \sigma^2 + \frac{1}{n^2} \sum_{i=1}^n \sum_{j \neq i} \text{cov}(D_i, D_j) \\
 &= \frac{\sigma^2}{n} + \frac{1}{n^2} \sum_{i=1}^n \sum_{j \neq i} \text{cov}(D_i, D_j)
 \end{aligned}$$

Hence if there are some positive dependencies (i.e., positive covariances) among the nearest-neighbor values (D_1, \dots, D_n) , then the second term of the last line will be positive, so that in this case $\text{var}(\bar{D}_n) > \sigma^2 / n$. Hence we must have

$$\begin{aligned}
 \text{(A1.4.2)} \quad E[(\bar{D}_n - \mu)^2] &> \frac{\sigma^2}{n} \Rightarrow \frac{n}{\sigma^2} E[(\bar{D}_n - \mu)^2] > 1 \Rightarrow E\left[\left(\frac{\bar{D}_n - \mu}{\sigma / \sqrt{n}}\right)^2\right] > 1 \\
 &\Rightarrow E(Z_n^2) > 1 \Rightarrow \text{var}(Z_n) > 1
 \end{aligned}$$

where the last line follows from the fact that $E(Z_n) = 0$ regardless of any dependencies among the nn-distances. But since one should have $\text{var}(Z_n) = 1$ under independent random sampling, it then follows that realized values of Z_n will tend to be farther away from zero than would be expected under independence. Thus even those clustering or uniformity effects due to pure chance will tend to look more significant than they actually are.

A1.5. The Point-in-Polygon Procedure

The determination whether a point, s , lies in a given polygon or not depends on certain basic trigonometric facts. In the Figure 1 below the (hollow) point s is seen to lie inside the polygon, R , determined by three boundary points $\{1,2,3\}$.

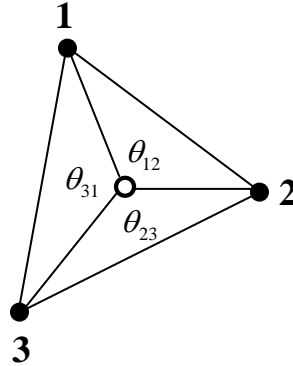


Fig.A1.1. Point Inside Polygon

If the angles (in radians) between successive points i and j are denoted by θ_{ij} , then it should be clear that for any point s inside R these angles constitute a full clockwise rotation through 2π radians, and hence that we must have $\theta_{12} + \theta_{23} + \theta_{31} = 2\pi$. The situation can be more complex when the given polygon is not convex. But nonetheless, it can easily be seen that if counterclockwise rotations are given negative values, then any counterclockwise rotations are canceled out by additional clockwise rotations to yield the same total, 2π . So if the polygon boundary points are numbered $\{1,2,\dots,N\}$ proceeding in a clockwise direction from any initial boundary point, then we must always have:¹

$$(A1.5.1) \quad \sum_{i=1}^{N-1} \theta_{i,i+1} = 2\pi$$

On the other hand, if point s is *outside* of the polygon, R , then by cumulating angles from s between each successive pair of points, the sum of clockwise and counterclockwise rotations must cancel, leaving a total of *zero* radians, i.e.,

$$(A1.5.2) \quad \sum_{i=1}^{N-1} \theta_{i,i+1} = 0$$

In the case of the simple polygon, $R = \{1,2,3\}$, above, this is illustrated by the three diagrams shown in Figure 2 below.

¹ Certain additional complications are discussed at the end of this section.

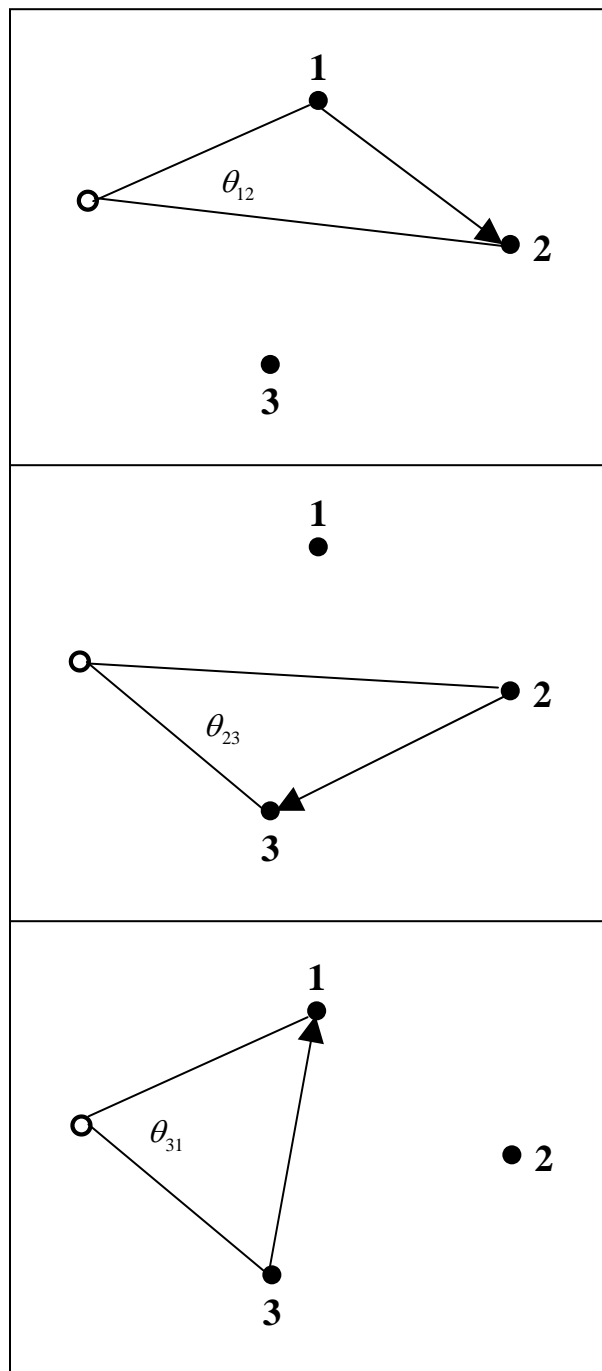


Fig.A1.2. Point Outside Polygon

Here the first two angles θ_{12} and θ_{23} are positive, and the angle θ_{31} is precisely the negative sum of θ_{12} and θ_{23} . By extending this idea, it is easy to see that a similar argument holds for larger polygons.

However, it is important to add here that this argument assumes that the polygon R is *connected*, and has *no holes*. Unfortunately, these conditions can sometimes fail to hold when analyzing general map regions. For example offshore islands are often included as part of larger mainland regions, creating disconnected polygons. Also certain small regions are sometimes nested in larger regions, creating holes in these regions. For example, military bases or Indian reservations within states are often given separate regional designations. There are other examples, such as the lake in Figure 2.4 of Part I, where one may wish to treat certain subregions as “holes”.

So when using standard point-in-polygon routines in practice, one must be careful to watch for these situations. Islands are usually best handled by redefining them as separate regions. Then by applying a point-in-polygon procedure to each region separately, one can determine whether a given point is one of them, or none of them. Holes can be handled similarly. For example if $R_1 \subset R_2$ so that the relevant region, R_2 , is given by the set-theoretic difference, $R_2 - R_1$. So for this region, one can apply point-in-polygon routines to R_1 and R_2 separately, and then accept only points that are in R_2 but not in R_1 .

A1.6. A Derivation of Ripley's Correction

First observe that the circular cell, C , of radius h about point s_i can be partitioned into a set of concentric rings, C_k about s_i , each of thickness Δ_k , so that $C = \bigcup_k C_k$. One such ring is shown in Figure 3 below.

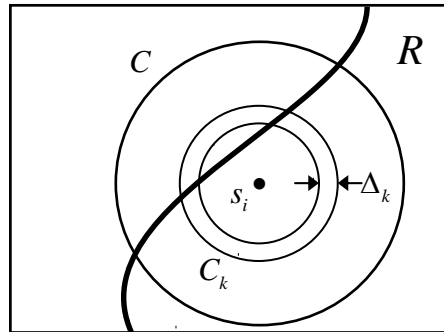


Fig.A1.3. Partition of Circular Cell, C

Since these rings are disjoint, it follows that the number of points in C is identically equal to the sum of the numbers of points in each ring C_k , so that (in terms of the notation in Section 2.2 in the text),

$$(A1.6.1) \quad E[N(C)] = \sum_k E[N(C_k)]$$

But by stationarity, it follows from expression (2.3.4) that

$$(A1.6.2) \quad E[N(C_k)] = \lambda a(C_k) = \lambda a(C_k \cap R) \left[\frac{a(C_k)}{a(C_k \cap R)} \right]$$

Where $a(C_k \cap R)$ is by definition the area of the observable portion of C_k inside R . Now when the ring thickness, Δ_k , becomes small, it should be clear from Figure A1.3 that the ratio of $a(C_k \cap R)$ to $a(C_k)$ is approximately equal to the fraction of the circumference of C_k that is inside region R . So if this ratio is now denoted by w_{ik} then,

$$(A1.6.3) \quad \frac{a(C_k \cap R)}{a(C_k)} \approx w_{ik} \Rightarrow \frac{a(C_k)}{a(C_k \cap R)} \approx \frac{1}{w_{ik}}$$

Hence, when the ring partition in Figure A1.3 becomes very fine, so that the Δ_k 's become small, one has the approximation

$$(A1.6.4) \quad E[N(C_k)] = \lambda a(C_k \cap R) \left[\frac{a(C_k)}{a(C_k \cap R)} \right] \\ = E[N(C_k \cap R)] \left[\frac{a(C_k)}{a(C_k \cap R)} \right] \approx \frac{E[N(C_k \cap R)]}{w_{ik}}$$

Putting these results together, we see that for fine partitions of C ,

$$(A1.6.5) \quad K(h) = \frac{1}{\lambda} E[N(C)] = \frac{1}{\lambda} \sum_k E[N(C_k)] \approx \frac{1}{\lambda} \sum_k \frac{E[N(C_k \cap R)]}{w_{ik}}$$

Note also that for sufficiently fine partitions it can be assumed that each ring contains *at most one* of the observed points, $s_j \in C \cap R$, so that the point-count estimators $\hat{E}[N(C_k \cap R)]$ for $E[N(C_k \cap R)]$ will have value one for those rings C_k containing a point and zero otherwise. Hence, observing by definition that $I_h(d_{ij}) = 1$ for all such points, it follows that

$$(A1.6.6) \quad \hat{E}[N(C_k \cap R)] = \begin{cases} I_h(d_{ij}) & , s_j \in C_k \cap R \\ 0 & , \text{otherwise} \end{cases}$$

If we again estimate λ by $\hat{\lambda} = n/a(R)$, and relabel the ring containing each point $s_j \in C \cap R$ as C_j , then (A1.6.6) is seen to yield the following estimate of $K(h)$ in (A1.6.5) based on point counts in the set $C \cap R$ centered at s_i ,

$$(A1.6.7) \quad \hat{K}_i(h) = \frac{1}{\hat{\lambda}} \sum_k \frac{\hat{E}[N(C_k \cap R)]}{w_{ik}} = \frac{1}{\hat{\lambda}} \sum_{j \neq i} \frac{I_h(d_{ij})}{w_{ij}}$$

Finally, by averaging these estimates over all points $s_i \in R$ as in the text, we obtain the pooled estimate,

$$(A1.6.8) \quad \hat{K}(h) = \frac{1}{n} \sum_{i=1}^n \hat{K}_i(h) = \frac{1}{\hat{\lambda}n} \sum_{i=1}^n \sum_{j \neq i} \frac{I_h(d_{ij})}{w_{ij}}$$

which is seen to be precisely *Ripley's correction* in expression (4.3.7).

A1.7. An Alternative Derivation of P-Values for K-functions

The text derivation of the P-values in expressions (4.6.8) and (4.6.10) is appealing from a conceptual viewpoint in that it focused directly on the distribution of the test statistic, $\hat{K}(h)$, under the CSR Hypothesis. But there is an alternative derivation of this expression that has certain practical advantages discussed below. This approach is actually much closer in spirit to the argument used in deriving the “envelope” P-values of expressions (4.6.3) and (4.6.4), which we now make more precise as follows. Observe that if l_0 is consistent with CSR then by construction (l_0, l_1, \dots, l_N) must be independently and identically distributed (*iid*) samples from a common distribution. In the envelope case it was then argued from the symmetry of *iid* samples that none is more likely to be the highest (or lowest) than any other. More generally, suppose we now ask how likely it is for the observed sample value, l_0 , to be the k^{th} largest among the $N+1$ samples (l_0, l_1, \dots, l_N) , i.e., to have *rank*, k , in the ordering of these values. Here it is important to note that ranks are not well defined in the case of ties. So for the moment we avoid this complication by assuming that there are *no ties*. In this case, observe that there must be $(N+1)!$ possible orderings of these *iid* samples, and again by symmetry, that each of these orderings must be equally likely. But since exactly $N!$ of these orderings have l_0 in the k^{th} position (where $N!$ is simple the number of ways of ranking the other values), it follows that if the random variable, R_0 , denotes the *rank* of l_0 , then under H_0 we must have:

$$(A17.1) \quad \Pr(R_0 = k) = \frac{N!}{(N+1)!} = \frac{N!}{(N+1) \cdot N!} = \frac{1}{N+1}, \quad k = 1, \dots, N+1$$

which in turn implies that the chance of a rank *as high as* k is given by,²

² Remember that “high” ranks mean *low* values of k .

$$(A1.7.2) \quad \Pr(R_0 \leq k) = \sum_{r=1}^k \Pr(R_0 = r) = \sum_{r=1}^k \left(\frac{1}{N+1} \right) = \frac{k}{N+1}, \quad k = 1, \dots, N+1$$

So rather than using the distribution of $\hat{K}(h)$ under CSR to test this null hypothesis, we can use the distribution of its rank R_0 in (A1.7.1) and (A1.7.2). But if we again let $m_+(l_0)$ denote the number of simulated samples at least as large as l_0 , then the observed rank of l_0 (assuming no ties) is precisely $m_+(l_0) + 1$. So to test the CSR Hypothesis we now ask: *How likely would it be to obtain an observed rank as high as $m_+(l_0) + 1$ if CSR were true?* Here the answer is given from (A1.7.2) by the *clustering P-value*:

$$(A1.7.3) \quad P_{cluster}(h) = \Pr[R_0 \leq m_+(l_0) + 1] = \frac{m_+(l_0) + 1}{N+1}$$

which is seen to be precisely the same as expression (4.6.8). However there is one important difference here, namely that we are no longer attempting to *estimate* a P-value. The distribution in (A1.7.1) and (A1.7.2) is *exact*, so that there is no need for a “hat” on $P_{cluster}$.

Another important advantage of this approach is that it is directly extendable to include possible *ties* among values. In particular, suppose that whenever two values are tied, we flip a fair coin to order them. More generally, suppose we use any tie-breaking procedure under which the rankings (R_0, R_1, \dots, R_N) are *exchangeable* random variables (i.e., under which their joint distribution is invariant under any permutation of the indices, $0, 1, \dots, N$). Then it again follows that all $(N+1)!$ orderings resulting from this procedure must be equally likely, and hence that (A1.7.1) and (A1.7.2) above continue to hold. Hence the key difference here is that in the presence of one or more ties, the ranking of l_0 is not uniquely determined by its value. There must be some additional tie-breaking procedure. So if l_0 is tied with exactly q of the simulated values, then there must be some additional information about the ranking, say $R_0(q)$, among these $q+1$ equal values. Hence all that can be said is that if $m_+(l_0)$ again has the same meaning then the final rank of l_0 will be $m_+(l_0) - q + R_0(q)$. For example, if l_0 were ranked last among the ties, so that $R_0(q) = q+1$, then l_0 would again have rank $m_+(l_0) - q + (q+1) = m_+(l_0) + 1$, since all tied values would be ranked ahead of l_0 (i.e., would be closer to rank 1 than l_0). Similarly, if l_0 were ranked ahead of all other ties, so that $R_0(q) = 1$, then l_0 would have rank $m_+(l_0) - q + 1$. Hence if we are given $R_0(q)$, then a *conditional cluster P-value* could be defined in terms of expression (A1.7.2) as follows:

$$(A1.7.4) \quad P_{cluster}[h | R_0(q)] = \Pr[R_0 \geq m_+(l_0) - q + R_0(q)] = \frac{m_+(l_0) - q + R_0(q)}{N+1}$$

But since the above exchangeability property also implies that

$$(A1.7.5) \quad \Pr[R_0(q) = i] = \frac{1}{q+1}, \quad i = 0, 1, \dots, q$$

it follows that we can obtain an *unconditional* clustering P-value (depending only on q) by simply taking summing out these conditioning effects as follows:

$$\begin{aligned} (A1.7.6) \quad P_{cluster}(h|q) &= \sum_{i=0}^q P_{cluster}[h | R_0(q)] P[R_0(q) = i] \\ &= \sum_{i=0}^q \frac{m_+(l_0) + 1 - i}{N+1} \left(\frac{1}{q+1} \right) = \frac{1}{(N+1)(q+1)} \sum_{i=0}^q [m_+(l_0) + 1 - i] \\ &= \frac{1}{(N+1)(q+1)} \left[\{m_+(l_0) + 1\}(q+1) - \sum_{i=0}^q i \right] \\ &= \frac{1}{(N+1)(q+1)} \left[\{m_+(l_0) + 1\}(q+1) - \frac{(q+1)q}{2} \right] \\ &= \frac{m_+(l_0) + 1 - (q/2)}{N+1} \end{aligned}$$

Hence this *generalized cluster P-value* amounts to replacing the rank, $m_+(l_0) + 1$, of l_0 in (A1.7.2) for the case of no ties with its *average rank*, $m_+(l_0) + 1 - q/2$, for cases where q values are tied with l_0 . So for example, if $N = 3$ and $(l_0, l_1, l_2, l_3) = (5, 2, 5, 6)$, so that $m_+(l_0) = 2$, $q = 1$ and the possible ranks of l_0 are $\{2, 3\}$, then its average rank is 2.5 and

$$(A1.7.7) \quad P_{cluster}(h) = \frac{(2+1) - 1/2}{5} = \frac{2.5}{5}$$

Note finally that the special case in (A1.7.3) above is now simply the special case of “no ties”, so that $P_{cluster}(h) = P_{cluster}(h|0)$.

The argument for uniform P-values is of course identical. Thus the corresponding *generalized uniform P-value* in the presence of q ties is given by:

$$(A1.7.8) \quad P_{uniform}(h|q) = \frac{m_-(l_0) + 1 - (q/2)}{N+1}$$

where $m_-(l_0)$ is again the number of simulated values l_i no larger than l_0 . Here it is important to note that these P-values are “almost complements” in the sense that for all q and h ,

$$(A1.7.9) \quad P_{cluster}(h|q) + P_{uniform}(h|q) = \frac{N+2}{N+1}$$

To see this, note simply that if we let $N_<, N_=: N_>$ denote the number of simulated samples that are less, equal, or greater than l_0 , then it follows by definition that $q = N_=:$, so that

$$(A1.7.10) \quad m_+(l_0) = N_> + N_=: N_> + q$$

$$(A1.7.11) \quad m_-(l_0) = N_< + N_=: N_< + q$$

and hence that

$$\begin{aligned} (A1.7.12) \quad P_{cluster}(h|q) + P_{uniform}(h|q) &= \frac{m_+(l_0)+1-(q/2)}{N+1} + \frac{m_-(l_0)+1-(q/2)}{N+1} \\ &= \frac{[(N_> + q)+1-(q/2)] + [(N_< + q)+1-(q/2)]}{N+1} \\ &= \frac{[(N_< + q + N_>)+2]}{N+1} = \frac{N+2}{N+1} \end{aligned}$$

Thus for even fairly small N it must be true that

$$(A1.7.13) \quad P_{cluster}(h|q) + P_{uniform}(h|q) \approx 1$$

so that we can essentially plot both P-values on one diagram. Hence all plots in K-function programs such as **k_function_plot** focus on *cluster P-values*, $P_{cluster}(h|q)$, where $P_{uniform}(h|q)$ is implicitly taken to be $1 - P_{cluster}(h|q)$.

A1.8. A Grid Plot Procedure in MATLAB

While the full grid, **ref**, can be represented in ARCMAP by exporting this grid from MATLAB and displaying it as a point file, it is often more useful to construct this display directly in MATLAB to obtain a quick check of whether or not the extent and grid size are appropriate. Assuming that the boundary file exists in the MATLAB workspace, this can be accomplished with the program **poly_plot.m**, which was written for this kind of application. In the present case the boundary file, **Bod_poly** (shown on page 3-23 of Part

I), is the desired input. Hence to plot the grid, **ref**, with respect to this boundary, use the command:

```
>> poly_plot(Bod_poly,ref);
```

The direct screen output of **poly_plot** is shown in Figure A1.4 on the right, and is clearly distorted relative to the true shape of Bodmin (since MATLAB assigns internal scales to each axis based solely on relative magnitudes).

To correct the shape, simply use the mouse to drag the bottom and sides of the frame until a more satisfactory image of Bodmin is obtained, such as that shown in Figure A1.5 below. This provides a very quick way of determining whether the grid extent and cell size looks appropriate for a reasonable interpolation to be done.

Notice also that the size of the dots in the Figure may be too large or too small, depending on the size of the boundary being used. These attributes (and others, such as the thickness of the boundary) can be altered. To do so, click on **Edit** and select **Current Object Properties**. Then to edit the size of the grid points, click on any of these points. You will then see that a few diagonal points are selected, and that a window has opened containing the attributes of these points. Observe that under “Marker” there is a point-type window and a numerical Marker size. If you increase or decrease this size, you will see that the point size in the display above has changed. In a similar manner, you can edit the boundary thickness by repeating the above **Edit** procedure, this time clicking on any exposed portion of the boundary, rather than on one of the grid points.

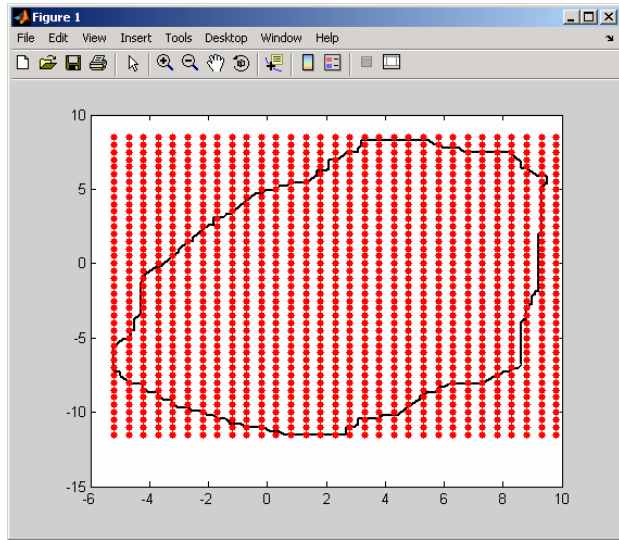


Fig.A1.4. Screen Output from **poly_plot**

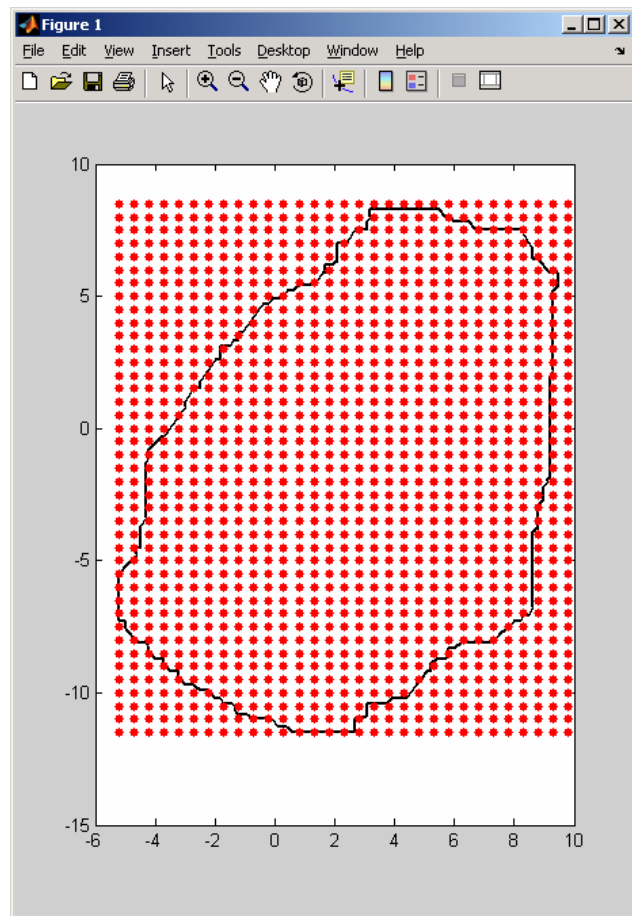


Fig.A1.5. Adjusted Screen Output

A1.9. A Procedure for Interpolating P-Values

To duplicate these results, open **Spatial Analyst** and then select:

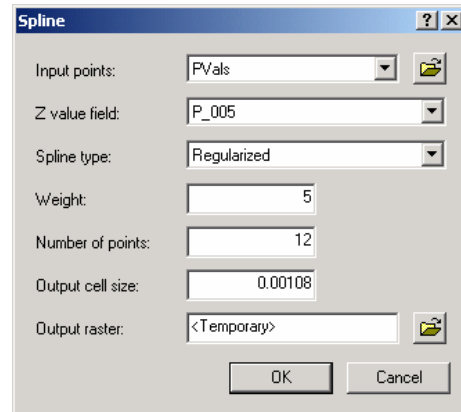
Interpolate to Raster → Spline.

In the **Spline** window that opens set:

Input points = “P-val.shp”

Z value field = “P_005”

Weight = “5”



and leave all other values as defaults. The value-field, **P_005**, contains the desired p-values in the file, **P-val.shp**. The weight **5** adds a degree of “stiffness” to the spline which yields a somewhat smoother result than the default **.01** value. Now click **OK** and a new layer appears called “Spline of P-val.shp”. Right click on this layer and select “Make Permanent”. Save it to your home directory as say, **spline_pvals**. This will not change the layer, but will give it an editable form. You can alter the display by right clicking on the layer, “Spline of P-val.shp”, selecting “Classified” (rather than “Stretched”), and editing its properties. [Notice that the values are mostly negative, and that the relevant range from 0 to 1 is only a very small portion of the values. This is due to the extreme nonlinearity of the spline fit.]

To obtain the display in Figure 4.23 above, this spline surface can be converted to contour lines as follows. First open **Spatial Analyst** again and this time select

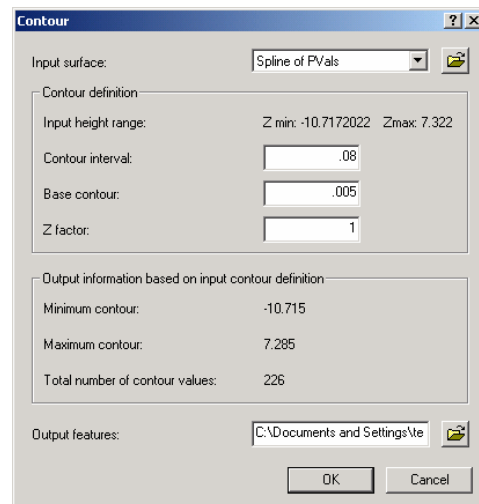
Surface Analysis → Contour

In the “Contour” window that opens set:

Input Surface = “Spline of PVals”

Contour Interval = “.08”

Base Contour = “.005”



Click **OK** and a new layer called “ctour” appears that shows the desired contours. This file is stored as a temporary file. You can edit its properties. So select “Classify” and choose the “Manual” option with settings (.01,.05,0.1,0.2) and appropriate colors. This should yield roughly the representation in Figure 4.23 above. This file is stored as a temporary file only. So you can keep trying different interval and base contour values until you find values that capture the desired regions of significance. Then use **Data → Export** to save a permanent copy in your home directory and edit as desired.