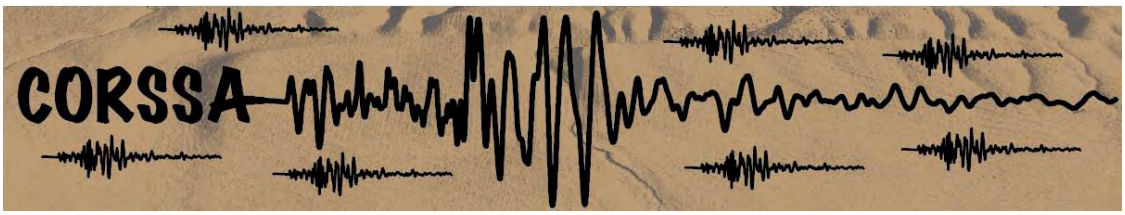

Theme III – Statistical Foundations

Exploratory Data Analysis: Magnitude, Space, and Time

Mark Naylor[1] • Katerina Orfanogiannaki[2] • David Harte[3]

1. School of GeoSciences, University of Edinburgh
2. National Observatory of Athens, Institute of Geodynamics
3. Statistics Research Associates

How to cite this article:

Naylor, M., K. Orfanogiannaki, and D. Harte (2010), Exploratory data analysis: magnitude, space, and time, Community Online Resource for Statistical Seismicity Analysis, doi:10.5078/corssa-92330203. Available at http://www.corssa.org.

Document Information:

Issue date: 16 November 2010 Version: 1.0

Contents

1 Motivation

This article will take you through an exploratory analysis of data contained in earthquake catalogues. The aim is to provide the reader with ideas about how to start investigating the properties of a new dataset in a straightforward and rigorous way. We start to introduce more advanced concepts, such as how to determine catalogue completeness, but reserve detailed descriptions of such advanced methodologies to other articles.

The target audience is undergraduate and graduate students who would like to use SSLib (Harte and Brownrigg 2010) and the R Language (Team 2010) to explore earthquake data. We have chosen to use R because it is freely available on all platforms and hope this makes the tutorial as accessible as possible. This article focusses on data exploration rather than being comprehensive guide to the application.

You will learn about basis plotting tools that can be used to explore the properties of earthquake data and visually identify difficulties in choosing a subset of the total catalogue for subsequent analysis. This article provides an introductory overview but does not provide technical solutions to those problems.

2 Starting Point

So, you have an earthquake catalogue... , most likely downloaded from a website or provided by a network. You may or (more likely) may not have been involved in producing the catalogue. Here we assume the latter.

Typically, the data in a catalogue consists of some measure of the size of an event (the magnitude), the plan view location of the event, the depth of the event and the time at which the event occurred. Catalogues represent earthquakes as having occurred instantaneously at a point in 3D called the hypocenter, the point where an earthquake originates. The epicentre is the point on the Earth's surface that is directly above the hypocenter.

Since the catalogue is a representation of "reality" it contains errors, artefacts and uncertainty. Some examples of these can be seen visually, demonstrations of which will be given in this article. Other catalogue problems require more a sophisticated quantitative analysis to identify. The analyses we present in this article are not sufficient to guarantee that catalogues are artefact free. Whenever a potential signal is derived from analysing a catalogue, we must work hard and critically to demonstrate that it is not an artefact.

We include snippets of code in order to provide the reader with a starting point for developing their own code rather than providing a comprehensive solution. The code snippets included in the text should run directly within R provided you have

R, SSLib and the associated libraries installed. Information about these libraries can be found in Appendix A.

For readers wanting to develop their own code in R and in need of a useful reference, we recommend the text by Crawley (2007).

3 Learning outcomes

After completing this article you should be able to perform an exploratory data analysis on different earthquake catalogues. The article will show you how to get a handle on the basic properties when presented with a new dataset. We facilitate access to datasets in different tectonic settings for the reader to explore as an exercise appropriate for undergraduate and graduate students.

4 Region choice

Earthquake catalogues cover extended regions which are often chosen as a matter of convenience, access and often representing political boundaries. In order to investigate a specific geophysical phenomenon, choosing an appropriate subregion motivated by the geology and tectonic regime is often required. However, the fractal nature of earthquake events makes appropriate spatial and temporal choice very difficult. What is the best way to place a Euclidean box or polygon around a non-Euclidean (fractal) cloud of epicentres? Do we focus on a major fault? A fault network? The whole catalogue? Also, the spatial extent of catalogues varies from a small geographic region, such as a single volcano, to global aggregates. When exploring data for new statistical phenomena we must be very careful that we do not introduce bias because of our criteria for sub-setting a catalogue; very subtle differences in 'choosing the box' can strongly affect the outcomes. It is always worth checking this selection afterwards to see if your conclusions are strongly affected by this, to a large extent, subjective choice.

In this article we will mainly focus on two specific regions of the North Island in New Zealand (Fig.1) to demonstrate the techniques, with one section looking at the 2004 Sumatra-Andaman earthquake. Details of how to access some other catalogues can be found in Appendix B.

The South Island is a collisional mountain belt accommodating oblique collision at a rate of 38mm/yr (Fig.2). The North Island is more volcanic.

The East Cape region, used in this article, lies in the north east of the North Island where the volcanic region trends back into a subduction zone heading north (Fig.2). The Wellington region lies to the southern end of the North Island (Fig.1).

Fig. 1 Map of New Zealand towns and cities; regions investigated in this article include the Wellington region around the Cook Strait and the East Cape region to the north of Gisborne.

5 Event magnitudes

A catalogue may contain magnitudes determined in different ways, e.g. from broad-band waveform inversion (seismic moment magnitude, M_w), narrow-band maximum amplitude data (local, surface or body wave magnitude M_L, M_S, m_b) or from calibrated historical data. Before starting any analysis it is important to research the

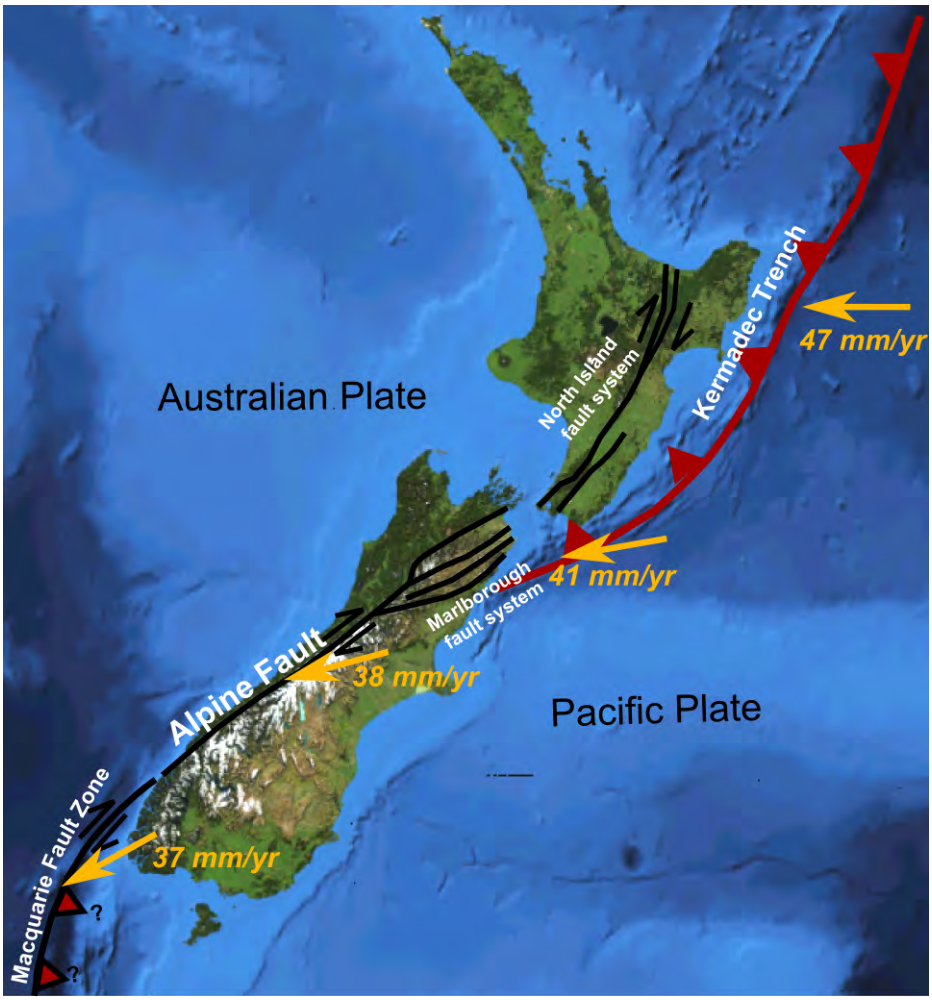

Fig. 2 Topographic map of New Zealand indicating the major fault systems, subduction zones, sense of motion and regional convergence rates.

way the catalogue has been constructed, finding out from the provider how homogeneous it is, how the network evolved to record smaller events more completely, and how the calibration was done. For more information on this see the Theme IV CORSSA articles by Woessner *et al.* (2010) and Gulia *et al.* (2010).

The events recorded in an earthquake catalogue represent the measurable subset of all earthquake events that occur in a region. Due to the spatial distribution of seismometers, there is not a spatially uniform ability to measure low magnitudes events.

5.1 Histogram of event magnitudes

A histogram of the magnitudes, counted in equal bins of size 0.2 magnitude units,is shown in Fig. 3a. The histogram contains no counts at magnitudes below the absolute sensitivity of the seismic network; this sensitivity decays away from individual seismometers and so is not spatially uniform resulting in the gradual rolloff at low magnitudes. Since the largest magnitude events are rare, there is a finite recorded maximum magnitude and the distribution also falls to zero at the high end. The region in the middle is a combination of complete data describing a real phenomenon and progressively incomplete data due to a combination of the spatial sensitivity of the seismic network, the depth distribution of the events and the local level of background seismic noise.

Frequency magnitude distributions are best analysed on a log-linear scale as in Fig. 3b. This shows clearly a linear relation, between magnitudes 4 and 7, known as the Gutenberg-Richter law.

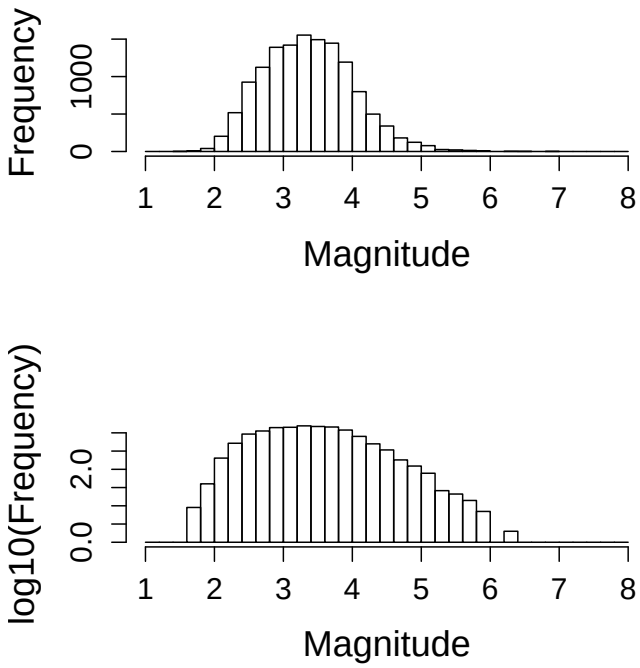

Fig. 3 Plot of the raw frequency-magnitude data in the East Cape region of New Zealand as a histogram with no filtering applied. Plotting the frequency data on a log axis is suggestive of a log-linear portion towards higher magnitudes indicating an exponential form. The code to produce these plots is in Table 1.

```
library(ssNZ)
a <- subsetcircle(NZ, minday=julian(1,1,1966), maxday=julian(1,1,2006),
                  minmag=1.0, maxradius=150, centrelat=-37.65,
centrelong=179.49)
as.catalogue(a, catname="EastCape")

b <- hist(EastCape$magnitude, breaks=seq(1,8,0.2), plot=FALSE)

# Plot the linear frequency histogram
par(mfrow=c(2,1))
plot(b, freq=TRUE, main="", xlab="Magnitude", ylab="Frequency", cex.lab=0.7,
cex.axis=0.6)

# Find positive frequencies
n <- which(b$counts>0)

# Take the log of the positive frequencies and replot
b$counts[n] <- log10(b$counts[n])
plot(b, freq=TRUE, main="", xlab="Magnitude", ylab="log10(Frequency)",
cex.lab=0.7, cex.axis=0.6)
```

Table 1 Code to plot raw histograms of the frequency magnitude on linear and log frequencies in Fig. 3.

Histograms record integer counts of individually measured events so the y-axis is not a continuous variable.The consequences of this are important and discussed in detail in later, for now we note the presence of 'empty' bins and a data scatter at high magnitudes.

5.2 Exponentially distributed earthquake magnitudes: Gutenberg-Richter

The frequency-magnitude of earthquakes can be displayed in two equivalent forms as an incremental or cumulative plot; historically the latter has been preferred. The use of cumulative data can easily be identified as the frequencies will always increase or stay the same from high to low magnitudes (Figs. 4a,c) . In contrast the incremental form just show the number of counts in each bin, where there are no events recorded in a bin it is left empty (Figs. 4b,d). The measure of frequency can be presented in different units such as counts (total number of events over some time period), rates (average number of events in some time period such as counts/yr) or as a probability (typically a normalised count to generate a probability density function).

Plotting the magnitude data against the logarithm of frequency suggests an exponential relation at intermediate to high magnitudes, as discussed in the previous section; this in referred to as the Gutenberg-Richter Law and is commonly expressed:

$$\log N(m_i > m) = -bm + a\,,$$

where N is the number of events with magnitude m_i greater than some threshold m and b is the log linear constant of proportionality referred to as the b-value. The constant a is just a function the total number of counts in the sample (or the rate etc. depending upon the y-axis units) at $m = 0$ in this cumulative form.

The b-value is an important parameter measured by seismologists because it tells us proportionately how many large and small events there are. A large value indicates more similar sized events are likely whilst a small value indicates a larger range of magnitudes.

5.2.1 Estimating the b-value: the first moment

We often wish to calculate the exponent relating earthquake frequencies and magnitudes, the b-value. There is a simple rigorous technique for doing this, however it is frequently done incorrectly and you need to be able to identify this.

The fitted b-value is frequently plotted on a cumulative frequency-magnitude Gutenberg-Richter plot (left hand side of Fig. 4). The cumulative frequencies are generated by starting at the highest observed magnitude and working back to the lowest, summing the frequencies as you go; this makes the frequency of each cumulative count dependent on all of the higher magnitude counts, and therefore introduces a correlation into the dataset. When the frequency-magnitude diagram is plotted, it is purely for illustrative purposes and as a check that the b-value is consistent with the data; no regression was performed on the cumulative data. In fact, regression should never be performed on cumulative data because the cumulative data is correlated to all higher magnitude events, making it fail the fundamental assumption that underlies basic regression that the data being modelled is uncorrelated.

Also, it is well known that the data must not be fitted using least squares because to do so provides a biased estimate of the b-value. The reason for this will be explained when we look at the variance in the frequency magnitude data at the end of Section 5.2.3.

A correct way to fit the data is to use the simple maximum likelihood technique derived by Aki (1965). This technique assumes both that the data is exponentially distributed and that the maximum magnitude is at infinity (unrealistic for an infinite sample; more likely is that the sample on which the b-value is being estimated is not large enough to sample the maximum magnitude), to derive the following relation for the b-value:

$$\widehat{b} = \frac{\log(e)}{\overline{m} - (m_{\min} - \Delta m/2)},$$

where $\overline{m}$ is the mean magnitude, $\widehat{b}$ is the estimate of b, $m_{\min}$ is the magnitude cutoff above which the catalogue is complete (Section 5.2.2), and $\Delta m/2$ is a correction for the finite binning width (Δm) of the original catalogue.

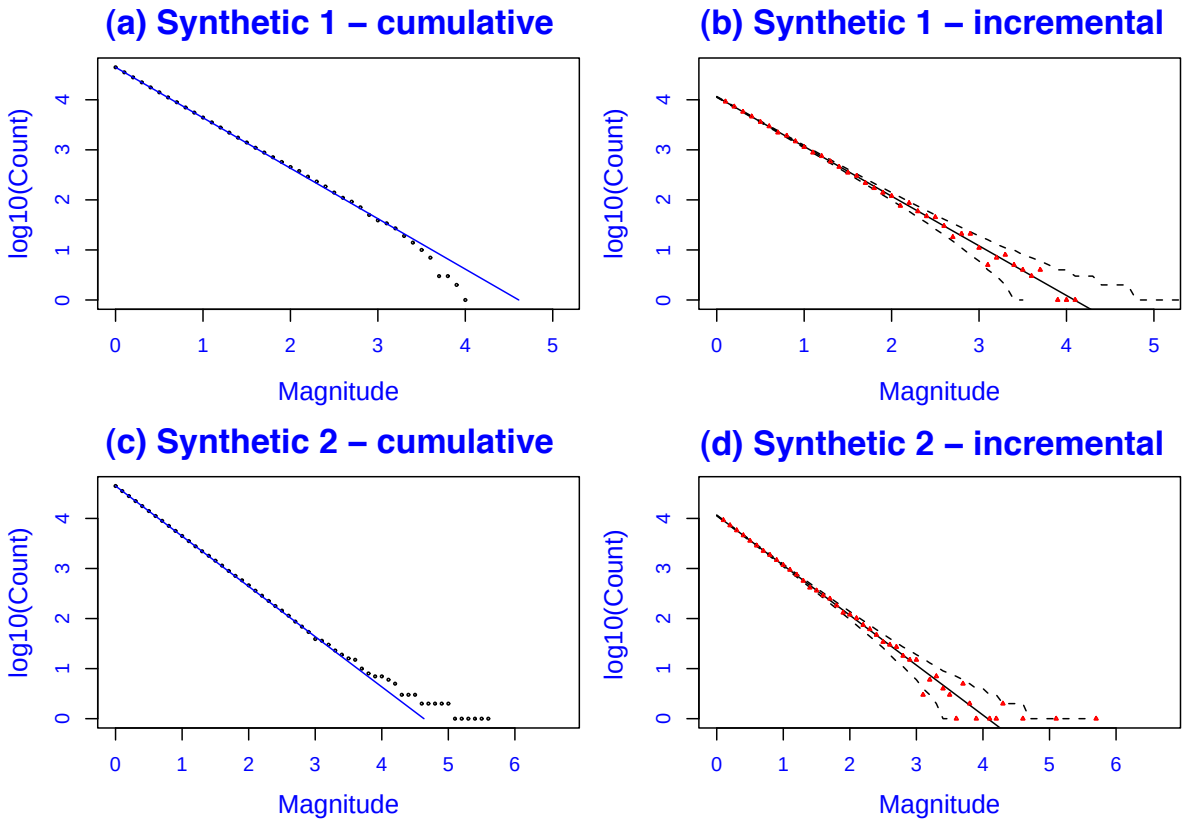

Fig. 4 Two synthetic samples (top and bottom rows) drawn from an exponential distribution plotted in cumulative form on the left and incremental form on the right with the log linear *b*-value slope added. In the cumulative form, one exhibits a rolloff (due to the chance under sampling of larger events) and the other a long tail (due to the chance sampling of larger events); they approximate typical end-member behaviours of exponential samples. 95% confidence intervals have been added to the incremental count data to illustrate the natural spread in exponentially distributed count data (the method to estimate and plot the confidence intervals is given later in Section 5.2.3, see Table 3 for the corresponding R code.)

Aki also shows that the uncertainty on this estimate can be estimated using:

$$\sigma_{\text{sterr}} = \widehat{b}/\sqrt{n}\,.$$

The standard error corresponds to an estimate of 1 standard deviation, i.e. a 68% confidence interval. Note that the standard error does not imply that the error is Gaussian distributed.

Calculating the *b*-value using this Maximum Likelihood method will always return an estimate of the *b*-value and an uncertainty estimate. However, the successful calculation of a *b*-value does not imply that the estimate is unbiased or that the data itself is exponentially distributed. When we choose to apply Aki, there is an

implicit assumption that we have decided that the data we are using is exponentially distributed, in other words, we are assuming that the data is described by an exponential model.

5.2.2 Catalogue completeness

Using a magnitude cutoff that does not lie above the completeness threshold for the catalogue will return a biased estimate of the b-value. The effect of completeness is investigated in Fig. 5 which compares the frequency-magnitude plot and the b-value estimate as a function of magnitude cutoff for two synthetic catalogues which are complete as a consequence of how they were generated and the East Cape catalogue, which we can clearly see deviates from a log-linear trend at low magnitudes likely as a consequence of incompleteness.

For the two synthetic catalogues, the horizontal line shows the exponent used to generate the synthetic catalogue; the more data there is, i.e. the lower the magnitude cutoff, the less biased and more accurate the estimate due to the reduction in the standard error. Note that despite there being large fluctuations in the tail, Aki's ML method still returns good estimates of the true b-value when there is sufficient data, at least 3 orders of magnitude are required for a reasonable estimate.

Unlike the synthetic catalogues, the b-value estimates in the East Cape catalogue diverge at low magnitude cutoffs, it is only stationary for intermediate values where there is sufficient data to obtain a good estimate but also where the data used in the estimate is complete, satisfying the implicit assumption that the data can described by an exponential model which is not the case at low magnitudes. The b-value is relatively stable above a cutoff of $m_c = 5$; the b-value estimate at $m_c = 5$ is 1.42 with a standard error of ± 0.12. Notice that the standard error does NOT quantify this bias and is not diagnostic of the correct model being assumed! Plotting the estimates as a function of the magnitude cutoff is a good way to demonstrate that a feature is robust to effects of completeness. These properties make the ML estimate a reasonable initial null hypothesis against which to test other models.

Exercise 5.1: Compare the magnitude ranges for the East Cape catalogue that provide the most reliable estimates with the frequency magnitude plot how easy is it to identify a completeness threshold? Choosing a completeness threshold by eye, especially on the cumulative plot, is dangerous and is unlikely to lead to a reliable and repeatable estimation of the b-value. See the CORSSA article by Mignan *et al.* (2010) for a discussion on the issues surrounding how to determine the completeness of a catalogue.

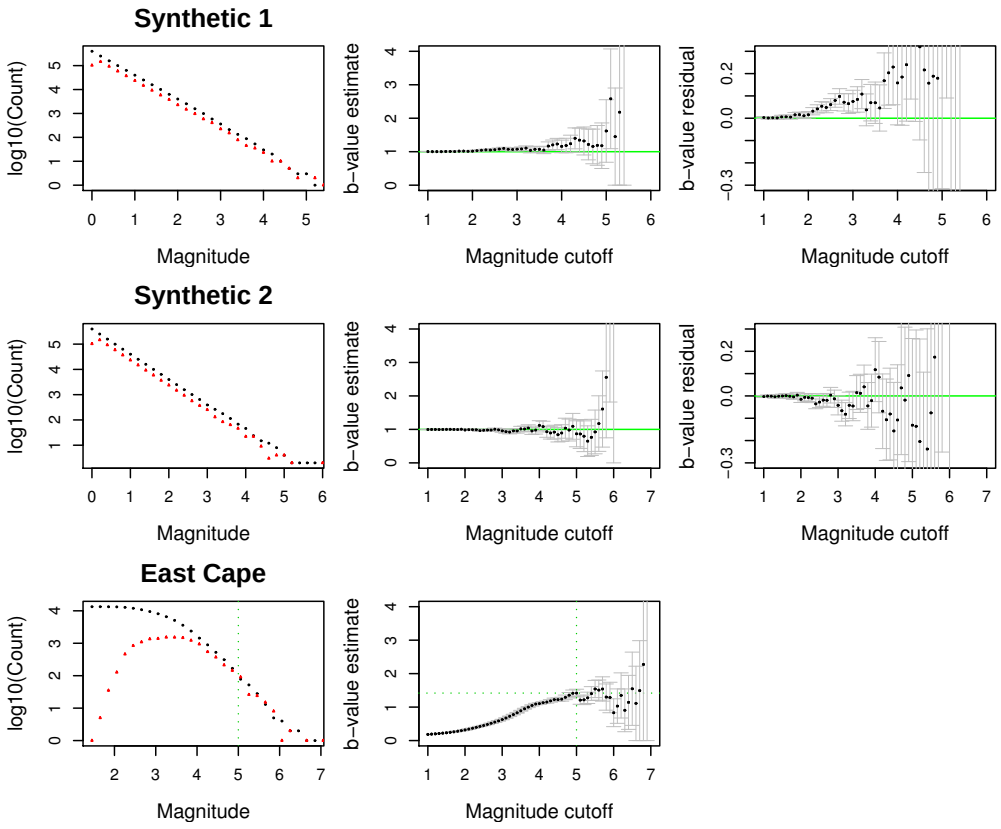

Fig. 5 Demonstration of the effect of magnitude cutoff on the estimation of the *b*-value using Aki's Maximum Likelihood technique. The columns show (1) the frequency magnitude plot with cumulative data in black and incremental data in red notice how incompleteness of the East Cape catalogue is evident in the declining counts at low magnitudes, (2) the estimated *b*-value with 1 standard error (68% confidence), for the synthetic cases the green line shows the known *b*-value used to generate the dataset and (3) for the synthetic cases, the residual from the known *b*-value as a function of the applied magnitude cutoff is shown. See Table 2 for the code for the *b*-value calculations.

5.2.3 Describing the statistical scatter

In order to interpret whether a frequency magnitude distribution contains more information than just the exponential relationship between frequency and magnitude, and to help remove the temptation of over-interpreting the data, you need to understand the inherent statistical scatter associated with sampling an exponential distribution.

Frequency magnitude plots record events taken from finite samples. The events in the tail of the distribution are probabilistically less likely; the difference in recording ~ 1 event makes a proportionately large difference to the observed counts. In contrast, at low magnitudes where there may be hundreds of events in each bin,

```
# Load New Zealand catalogue
library(ssNZ)

# Define a function to plot vertical error bars
error.bar <- function(x, y, upper, lower=upper, length=0.05){
    arrows(x,y+upper, x, y-lower, angle=90, code=3, length=length, lwd=0.5)
}

# Define a catalogue subset and extract magnitudes
a <- subsetcircle(NZ, minday=julian(1,1,1966), maxday=julian(1,1,2006),
minmag=1.0,
                  maxradius=150, centrelat=-37.65, centrelong=179.49)
as.catalogue(a, catname="EastCape")
Mi <- EastCape$magnitude
maxmag <- ceiling(max(Mi))

# Set list of magnitude cutoffs
incr <- 0.1
Mc <- seq(1, maxmag, incr)

# Loop over all magnitude cutoffs to calculate
b-value, standard error and save
to ANS
ans <- NULL
for (i in 1:length(Mc)) {
    j <- which(Mi>=Mc[i])
    rate.mle <- 1/mean(Mi[j] - Mc[i])
    b.mle <- rate.mle/log(10)
    b.sterr <- b.mle/sqrt(length(j))
    ans <- rbind(ans, c(Mc[i], b.mle, b.sterr))
}
colnames(ans) <- c("Mc", "b", "sterr")

# Plot b-value vs magnitude cutoff with error bars
plot(ans[,"Mc"], ans[,"b"], xlab="Mc", ylab="b-value est.", ylim=c(0,4),
cex=0.2)

error.bar(ans[,"Mc"], ans[,"b"], ans[,"sterr"])
```

Table 2 Code to calculate the b-value and uncertainty as a function of magnitude cutoff, see Fig. 5.

the difference in observing events on ~ 1 has a proportionately small effect. This difference is further accentuated by the use of log y-axis.

We need to understand the scatter in the frequency-magnitude data about the exponential trend because this tells us about the amount of scatter we should expect to observe when we look at data drawn from an exponential distribution.

The easiest way to demonstrate these fluctuations graphically is to draw a large number of samples of n events from an exponential distribution with a constant

exponent (i.e. b-value). Fig. 6a,b shows the cumulative and incremental frequency-magnitude plot for 500 samples of 50,000 events each drawn from an exponential distribution with $b = 1$. Notice the wide spread in the counts at high magnitudes since we have generated this plot by sampling the exponential distribution directly, this is the statistical noise we should expect to see in real earthquake frequency-magnitude data just because of sampling effects. The spread of the bounds on the cumulative and incremental data can be compared in Fig. 6c, note that the gradient at low magnitudes is the same but the curves are offset from each other.

Fig. 6d shows one of the 500 samples and adds confidence intervals calculated using Poisson counting errors (see below for method). We can see that these calculated confidence intervals approximate the spread shown in the stacked synthetic data and that as the probability increases they tend to each other. The confidence intervals allow us to visually assess whether a sample is unlikely to be drawn from an exponential distribution (and motivating an investigation of models other than the Gutenberg-Richter Law).

Fig. 4 showed two synthetic samples plotted separately in cumulative and incremental form; in the first sample there is a chance deficit of large events and in the second several large events were sampled - both consistent with being drawn from the exponential distribution.

The first cumulative sample shows an apparent roll off at high magnitudes. All real catalogues must show a real rolloff in the frequency magnitude data controlled by the largest possible earthquake in a region, and ultimately limited by the size of the Earth. However, just because we see a rolloff in the data does not imply we have observed the largest possible earthquake yet - a comprehensive sample may in fact take $>$ 1000 yrs to observe directly. The 95% confidence intervals plotted on the incremental data confirms that this sample does indeed lie well within the statistical noise; we would require a very sharp rolloff to demonstrate that it lies outwith this natural statistical noise.

The second sample demonstrates the bias in targeting a catalogue known to contain a large earthquake. Whilst the tail of this diagram does not lie on the trend line it does still lie within the confidence intervals on the incremental data. Whenever a catalogue is chosen because it contains a particularly significant event; a selection bias is introduced that will make the frequency magnitude plots look more like Fig. 4 because of that(those) large event(s). When statistically analysing data, it is crucial that you are conscious of the possibility of artificially generating a signal because of data selection criteria introducing a selection bias. Sometimes features like this have been over interpreted as statistical outliers to support the interpretation of the presence of earthquakes with a characteristic size.

Exercise 5.2: Convince yourself that the cumulative plot is derived from the data in the incremental plot. Explain the relationship between the incremental and

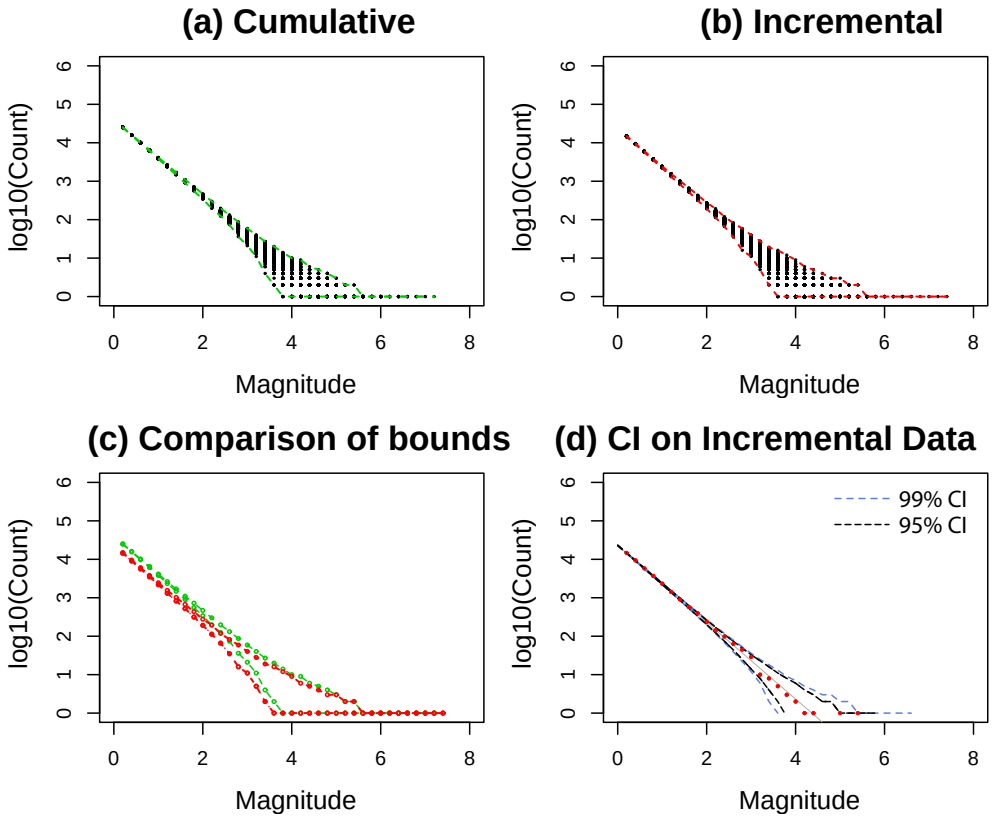

Fig. 6 Exploration of the statistical scatter in frequency-magnitude data. 500 samples of 50,000 events have been processed to generate 500 superimposed frequency-magnitude plots in (a) cumulative form (b) incremental form and (c) with the upper and lower bounds extracted from (a) and (b) for comparison. (d) shows a single sample plotted in incremental form with the calculated 95% (black dashed) and 99% (blue dashed) Poisson confidence intervals added; note how they capture the form of the natural spread shown in (b) and that as the percentage confidence increases they tend to the upper and lower bounds.

cumulative plots in Fig. 4 and 6. What feature of the frequency magnitude data makes the cumulative counts sit systematically above the b-value in the first synthetic and systematically below the b-value in the second synthetic?

Given some sample of magnitudes, we might want to ask whether the distribution is consistent with being drawn purely from an exponential; if it is we should expect the statistical scatter of the data to lie within a range similar to that presented in Fig. 6. Luckily, we can estimate confidence intervals on count data using the Poisson distribution (added to the incremental data in Figs. 4b,d and 6d) which is a discrete probability distribution that expresses the probability of a number of events occurring in a fixed period of time if these events occur with a known average rate and independently of the time since the last event. Code to generate and plot

the confidence intervals is given in Table 3. The rate and the b-value for a given exponential distribution are linearly related and the 95% confidence intervals lie between the 2.5% and 97.5% quantiles.

We now demonstrate how to calculate appropriate confidence intervals for frequency-magnitude data presented in a histogram. For the integer counts that define the frequency, the Poisson distribution provides a good approximation to the 'statistics of small numbers' for the rare, large events. The central limit theorem implies that it also converges to a narrower Gaussian distribution in the limit of very large numbers for the better sampled smaller events. This means it matches the 'trumpet-like' scatter in the multiple synthetic realisations on Fig. 6 well, both at small and large magnitudes.

The density function of the Poisson shows the probability of obtaining a count of x when the mean count per unit is λ:

$$p(x) = \frac{\lambda^x}{x!} \exp(-\lambda) \, .$$

However, this is not enough on its own to calculate the confidence interval on a given count for a particular magnitude. In probability and statistics, uncertainties are generally determined by specifying the 'quantile function' of the probability distribution of a random variable. This specifies the value of the random variable below which a certain fraction of samples will fall, on average; it is useful when users need to know key percentage points of a given distribution, for example the upper and lower 2.5%, with 95% confidence being between these two points. We now show how to do this in practice.

The Poisson quantile function in R, $qpois(p, \lambda)$ takes the mean count λ, and uses the equation above to estimate how many counts correspond to the p^{th} confidence interval. For example if $p = 0.9$ and the mean count is 20, $qpois(0.9, 20) = 26$. This tells you that the average of many random samples will have 90% of the sample sizes less than or equal to 26. In the code presented in is given in Table 3, the mean count, C, is estimated using the maximum likelihood fit to the data. The 95% confidence intervals are estimated using the 2.5% and 97.5% quantiles.

Further, we can now understand why the least squares fitting technique generates a biased estimate of the b-value. The reason a linear least squares fitting should not be used on the raw frequency-magnitude data is because (i) not all of the points carry the same weight the points in the tail correspond to a single event whereas the points at lower magnitudes contain potentially hundreds of events and the significance of these points needs to be weighted accordingly and (ii) the variance is not constant and it is asymmetrically distributed at high magnitudes and therefore not Gaussian distributed. The large fluctuations in the tail would have a disproportionate effect on the gradient of the line given that they arise from a low proportion of observations.

```
# Sets parameters
bvalue <- 1
rate <- bvalue*log(10)
N <- 50000
delta <- 0.1

# Generate syntheic catalogue
Mi <- rexp(n=N, rate=rate)
Mc <- 0

# Calculate histogram
Z <- hist(Mi, plot=FALSE, breaks=seq(min(Mi), max(Mi+2), delta))

# Estimate rate and b-value using ML
rate.mle <- 1/mean(Mi - Mc)
b.mle <- rate.mle/log(10)

# Plot data with counts > 0
m <- which( Z$counts>0 )
plot(Z$mids[m], log10(Z$counts[m]) , pch=2, col=2, cex=0.4, xlab="Magnitude",
     ylab="log10(Count)", xlim=c(0,max(Z$mids)))

# Plot ML estimate
C <- dexp( Z$mids, rate.mle) *delta*N
lines(Z$mids, log10(C), lty=1)

# Add 95 % confidence intervals
qhi <- 0.975
qlo <- 0.025

nhi <- qpois(qhi,C)
nlo <- qpois(qlo,C)

points(Z$mids, log10(nhi), lty=2, type="l")
points(Z$mids, log10(nlo), lty=2, type="l")
```

Table 3 Code to generate a synthetic catalogue, plot as incremental frequencies with the ML best fit line and 95% confidence intervals added (See right column in Fig. 4).

6 Simple epicentral plot

An epicenter plot shows the latitude-longitude position of each event in map view (Fig. 7). It is immediately evident is that the events are not uniformly spaced; the data naturally clusters. Having seen this spatial clustering, a natural question to ask next is whether there exist correlations between the epicenters and clustering in time, magnitude or depth. This plot is very basic cloud of points; we develop this plot into something more useful in the following sections.

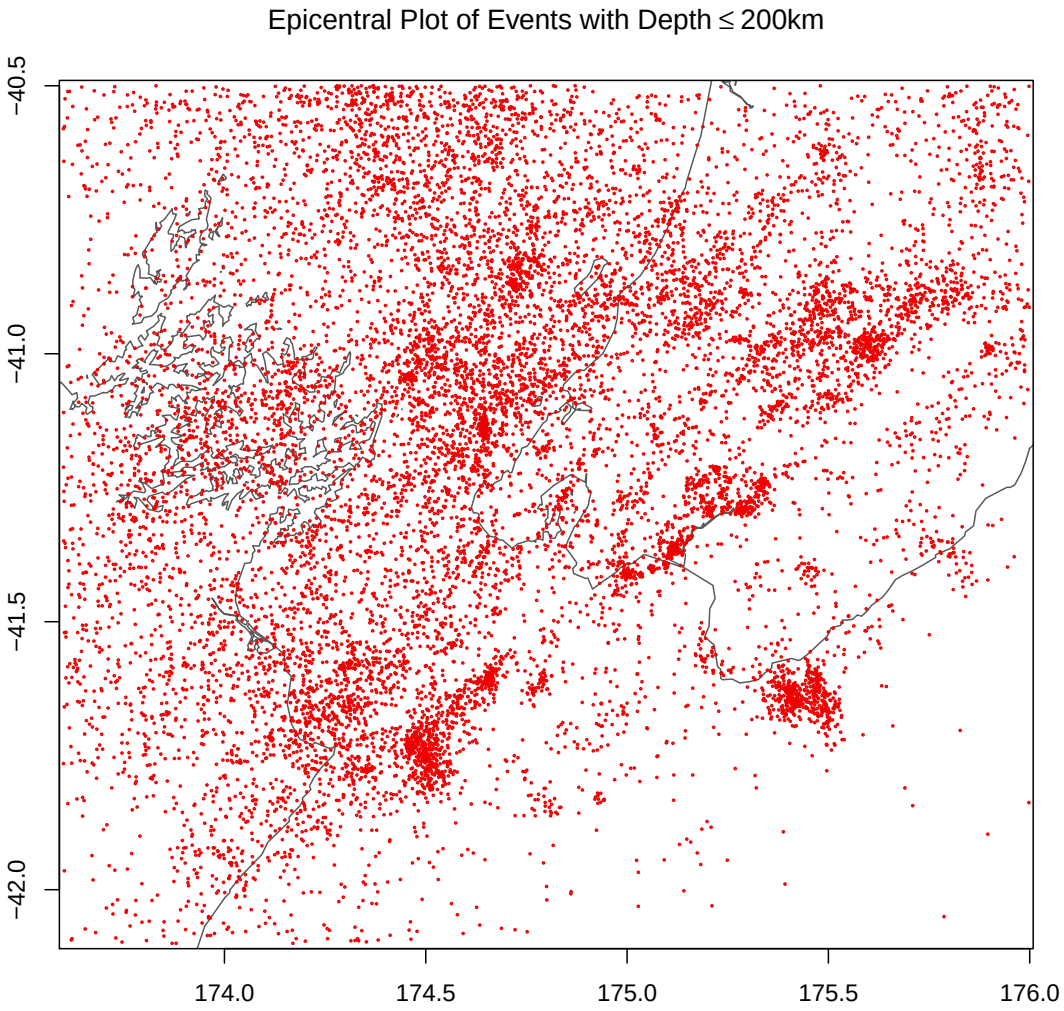

Fig. 7 Plot of earthquake epicenters on a basemap of the Wellington region. The x and y axis show the latitude and longitude; red points show the event locations and the black lines show the coastline. The R code to produce this plot is shown in Table 4.

function requires both GGobi and the rggobi package to be installed. maxlong=176.0, minday=julian(1,1,1978)), catname="Wellington") maxday=julian(1,1,1992), maxdepth=200)

```
library(ssEDA)
library(ssNZ)

#   use high resolution map if mapdata package available
if (require(mapdata)) mapnm <- "nzHires" else mapnm <- "nz"

b <- subsetrect(NZ, minmag=2, maxdepth=200,
                minday=julian(1,1,1978), maxday=julian(1,1,1992),
                minlat=-42.1, maxlat=-40.5, minlong=173.6, maxlong=176.0)

epicentres(b, criteria=FALSE, mapname=mapnm)
title(main=expression(paste("Epicentral Plot of Events with ", Depth <= 200,
"km")))
```

Table 4 Code to make a basic epicentral plot showing only the location of events on a coarse scale map projection, see Fig. 7.

```
library(ssEDA)
library(ssNZ)

#   use high resolution map if mapdata package available
if (require(mapdata)) mapnm <- "nzHires" else mapnm <- "nz"

b <- subsetrect(NZ, minmag=2, maxdepth=200,
                minday=julian(1,1,1978), maxday=julian(1,1,1992),
                minlat=-42.1, maxlat=-40.5, minlong=173.6, maxlong=176.0)

epicentres(b, usr=c(173.55, 176.05, -42.13, -40.47), depth=c(0, 30, 50, 70, 100,
Inf),
           criteria=FALSE, magnitude=c(2, 3, 4, 5, 6, Inf), mapname=mapnm)
title(main=expression(paste("Epicentral Plot of Events with ", Depth <= 200,
"km")))
```

Table 5 Code to create an epicentral plot where the colour indicates depth and the size indicates the magnitude of the events, see Fig. 8.

7 Enhancing the epicentre plot with magnitude and depth

This time we enhance the epicentral plot by varying the colour and size of each event using the depth and magnitude marks respectively. We can see a trend from warm colours in the south east to cold colours in the north west indicating progressively deeper event to the north west. There are no deep events in the south east but some shallow events do occur in the north west.

Exercise 7.1: Locate the epicenter plot (Fig. 8) on the maps of New Zealand shown in Fig. 1 and 2. Sketch a cross-section running from the south east to the

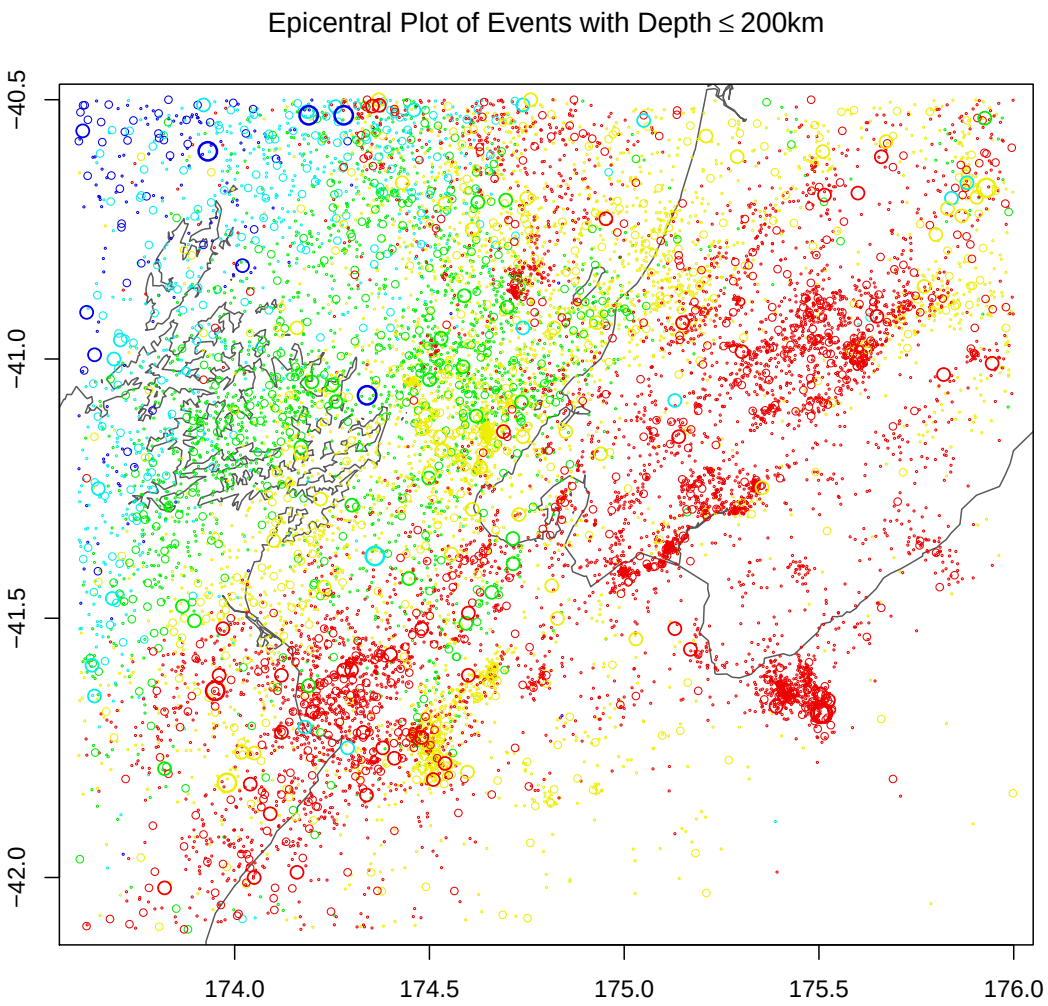

Fig. 8 Plot of earthquake epicenters on a basemap coloured by depth and with size proportional to the event magnitude. Warm colours indicate shallow depths and cool colours deeper events in intervals of Depth, Z [km]: 30 $\leq Z$ red ; 50 $\leq Z$ yellow ; 70 $\leq Z$ green ; 100 $\leq Z$ cyan ; $\infty \leq Z$ blue. The symbol size steps at integer magnitude boundaries from 2 upwards. The R code to produce this plot is shown in Table 5.

north west. Can you think of a geological process that might be consistent with this geometry?

Exercise 7.2: Using one of the catalogues described in Appendix B, generate an epicentral plot enhanced with magnitude and depth by modifying the code snippet in Table 5. What trends can you see in the data? Can you suggest what processes generate these patterns?

```
library(ssEDA)
library(ssNZ)

b <- subsetrect(NZ, minmag=2, minday=julian(1,1,1978), maxday=julian(1,1,1992),
maxdepth=200,
                minlat=-42.1, maxlat=-40.5, minlong=173.6, maxlong=176.0)

rotation(b, theta=-40)
title(main="Plate Subduction in Wellington Region")
```

Table 6 Code to plot a vertical cross section of events, see Fig. 9.

8 Depth cross-section

Having identified the depth trend in Fig. 8 we can explore it in more detail by taking a cross-section perpendicular to the direction of dip in a north west to south east direction (Fig. 9). We can see that the zone of seismicity is broad and dips most steeply to the north-west. This feature arises because the Wellington earthquake dataset is located over the subduction zone in New Zealand; the depth of the events is describing the structure of the subducting slab. The geometry of the subducting slab is clearly defined with events extending down approximately 100–200 km, into the mantle. Note the horizontal sheets of events appear to occur at particular depths in the upper part.

Question: At least two distinct horizontal trends in event location can be seen in the data at depths below 50m. Can you see them? Do you think it is likely that they are real or not? Why? We will investigate these further in the next section.

Exercise 8.1: In the last exercise you used and epicentral plot to describe spatial-depth-magnitude trends. Now use the code in Table 6 to generate interesting cross-sections through the epicentral plot to support your description of the data. Experiment with cross-sections at different angles. What new features can you see? Are there any trends that may be artefacts?

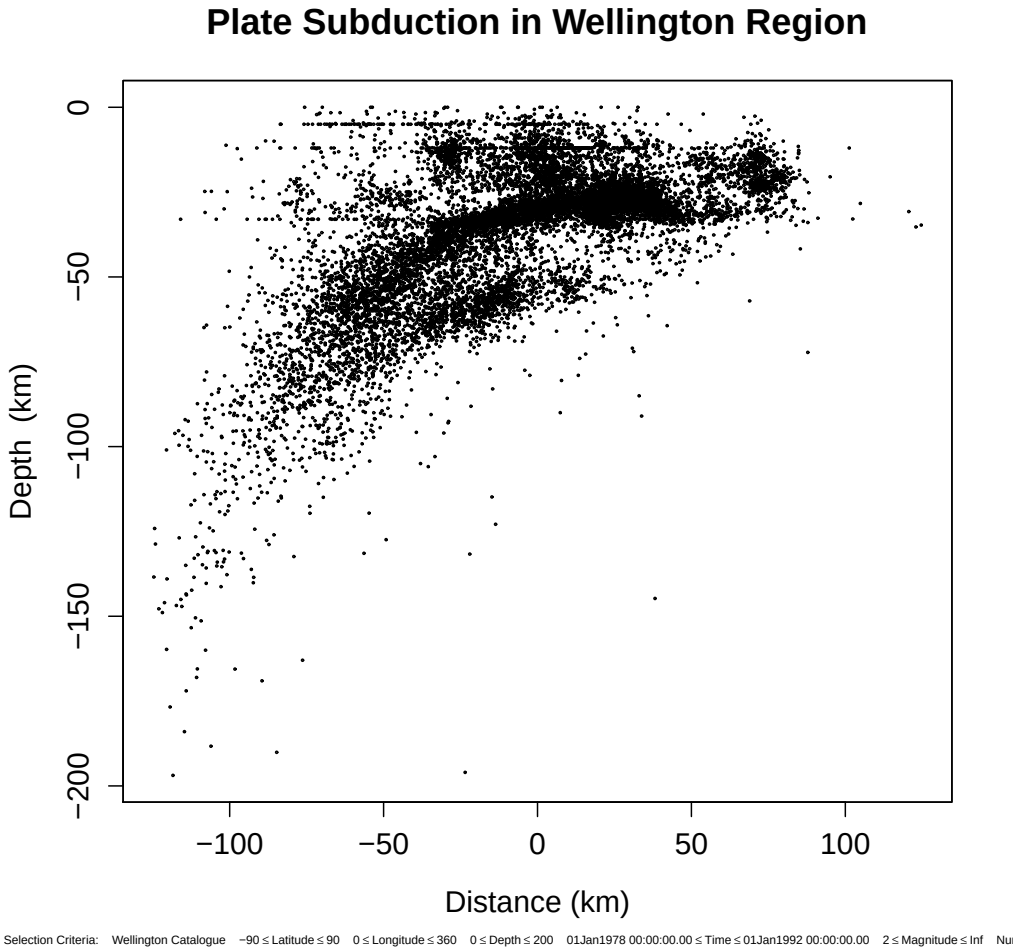

Fig. 9 Depth cross section through the New Zealand subduction zone plotting individual earthquake events as points. The position of the events illuminate the geometry of the subducting and overriding slabs. See the corresponding R code in Table 6.

9 Event depths

This histogram has been created from a catalogue subset containing ~ 5000 events. The deep earthquakes show a relatively smooth depth distribution of events. In contrast, events at shallow depths appear much more punctuated.

At shallow depths the majority of events are contained in bins ending at 12km, 32km, 5km and 20km (largest count to smallest count respectively). The occupation of these four bins is substantially greater than any others and they contain most of the events. They seem at first glance to represent sheets of events at these

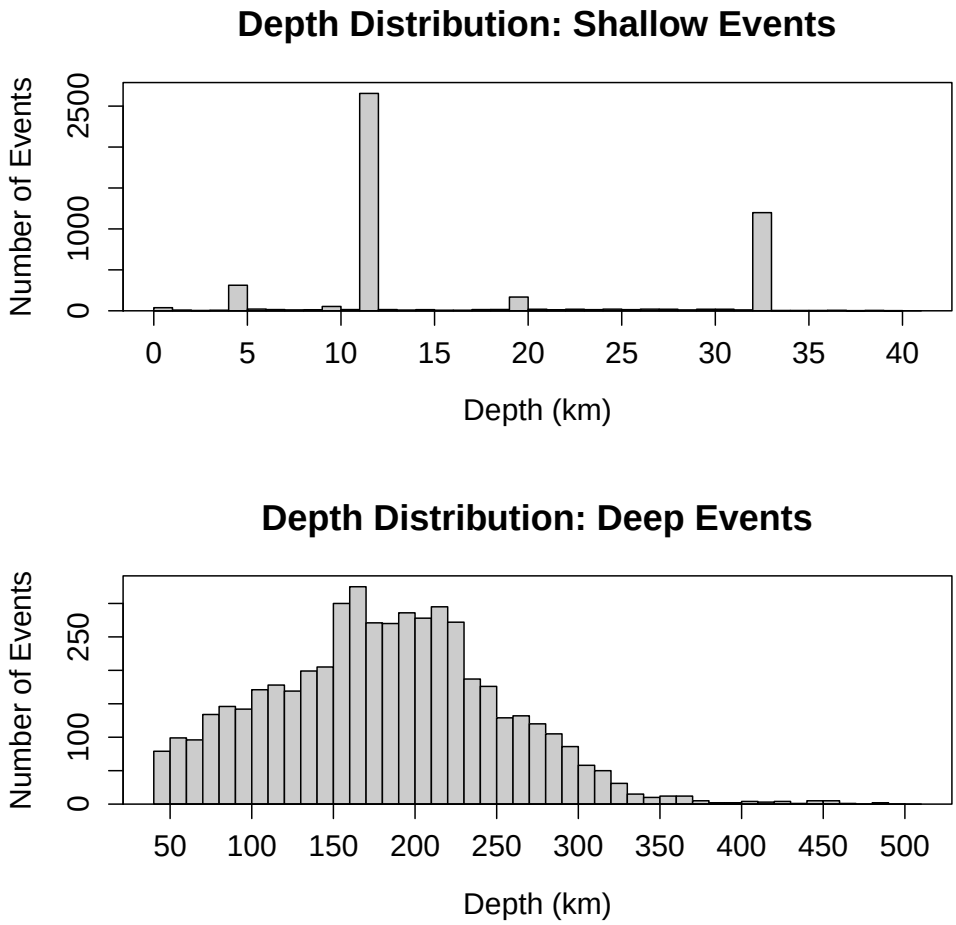

Fig. 10 Histogram of the depths of earthquake events recorded in the catalogue for (a) shallow and (b) deep events. See Table 7 for the corresponding R code.

depths located within ±500m. Given that this appears a significant trend, we must ask whether it is a real, physical feature where these depths represent the actual depths at which earthquakes occur an imply some geologically interesting feature OR whether this is actually an artefact contained in the earthquake catalogue. Communication with those people who generate the earthquake catalogues can prove invaluable here. A single question to ask is what is the starting depth in the iteration for the hypocentre inversion. Here it looks as if 12km and 33km have been used. The fact that the starting depths have not moved implies that they are not real, essentially the depth is undetermined for these events.

```
library(ssEDA)
library(ssNZ)

a <- subsetrect(NZ, minday=julian(1,1,1965), maxday=julian(1,1,1995),
                mindepth=0, maxdepth=39.99, minmag=4)

depth.hist(a)
title(main="Depth Distribution: Shallow Events")
```

Table 7 Code to plot a histogram of the depths of events in a catalogue subset using the SSLib function, see Fig. 10.

10 Space-time clustering at the Andaman Islands

In this section we move away from the New Zealand example used in the rest of this article to briefly look at the Andaman Islands earthquake and its aftermath.

The 2004 Boxing Day earthquake in the Indian Ocean was an earthquake with magnitude between 9.1 and 9.3, it is the second largest to be recorded on a seismograph and is known as the Sumatra-Andaman earthquake. Since then there have been several large events. In this section, we want to investigate spatial-temporal clustering of successive large events.

10.1 Enhancing the epicentre plot with time

The basic epicentre plot only showed that there was some spatial structure. We can gain much more insight by decorating the points using other information (marks). First we use a temporal mark (Fig. 11) to colour events within 30 days of the five largest events. The size of the circles indicates the magnitude of the events. We can immediately see that the events are localised in time as well as space and that these "hotspots" of activity occur in different parts at different times. In this example, the earthquake events delimit the plan view extent of the subduction zone.

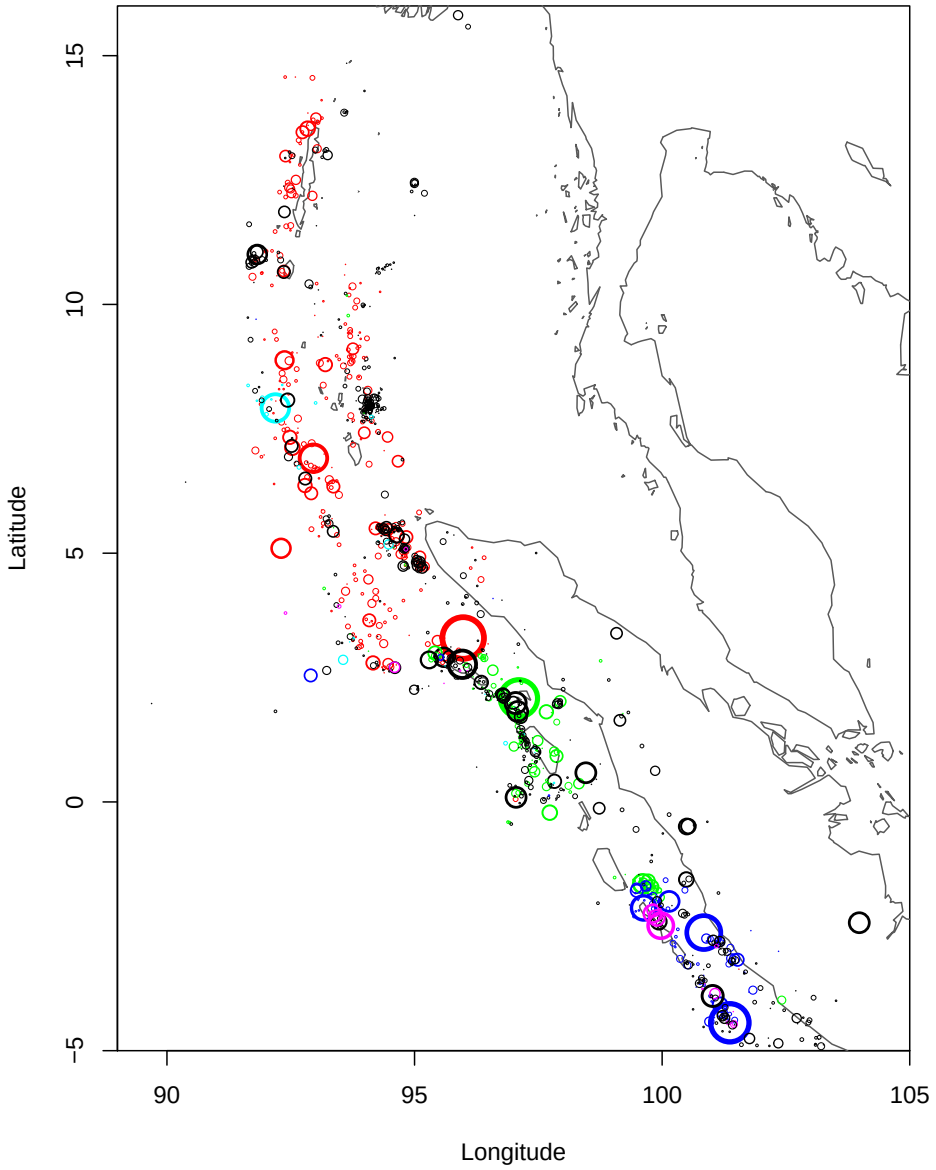

Fig. 11 Plot of earthquake epicenters around the region where the Sumatra-Andaman earthquake occurred. Several large events have been picked out and the events within a 30 day window coloured to highlight the spatial and temporal clustering. See the corresponding R code in Table 8.

```
# Load libraries and data
library(PtProcess)
library(mapdata)
data(Phuket)

# Choose Big Events
big <- list()
big$time <- c(360.040, 452.673, 570.654, 1350.465, 1516.358)
big$date <- c("26Dec04", "28Mar05", "24Jul05", "12Sep07", "20Feb08")

# Required graphics parameters
usr <- c(89, 105, -5, 16)
mai <- c(0.85, 0.85, 0.05, 0.15)
size <- Phuket$magnitude - 4.95

plot.new()
par(mai=mai, usr=usr)
title(xlab="Longitude", ylab="Latitude")
axis(1)
axis(2)
box()
map("world2Hires", add=TRUE, interior=FALSE, col="gray35")

# Colour if within 30 days of one of 5 major events, else black
for (j in 1:length(Phuket$latitude)) {
    col <- 1
    for(k in 1:5) {
        if((Phuket$time[j] > big$time[k]) & (Phuket$time[j] < big$time[k]+30))
            col <- k+1
    }
    points(Phuket$longitude[j], Phuket$latitude[j], cex=size[j], col=col,
           lwd=size[j], pch=1)
}
```

Table 8 Code to generate the epicentral plot of the Sumatra region with events coloured as 30 day temporal clusters of events after one of 5 large events, see Fig. 11. Events in black fall outwith one of these 30 day windows.

10.2 Latitude-time plot

Fig. 12 colours the largest events and those events in a 30 day window after the main event. The latitude time plot nicely shows how events cluster in space and time. In this case the latitude is a good separator since the subduction zone is orientated approximately North-South. For a more angles trend, it would be necessary to rotate the axes parallel to the strike of the zone.

Question: Using Fig. 11 and 12 explore the following questions.

Why does the latitude time plot successfully pick out the spatial clustering of events along Sumatra? Why would a longitude time plot not be as useful?

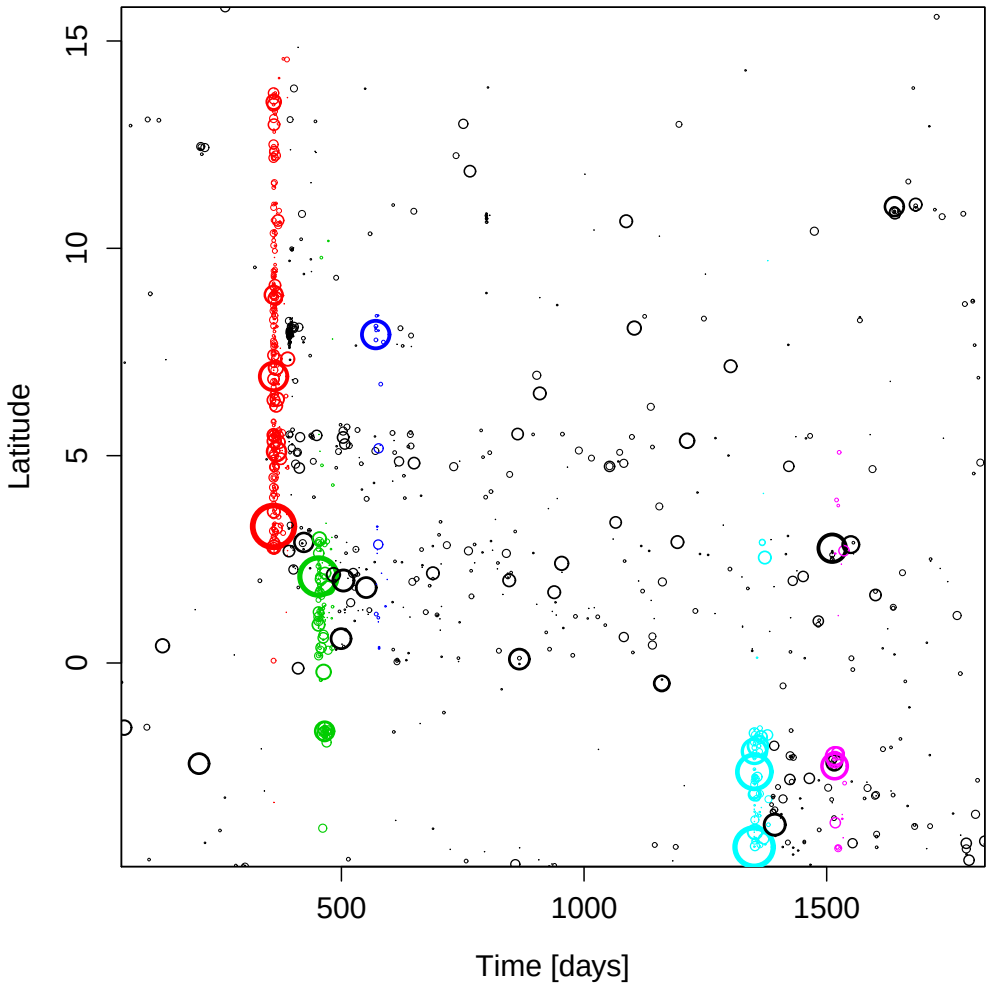

Fig. 12 Plot of the latitudinal distribution of events with time coloured in 30 day periods from the five large events and with the size proportional to the magnitude of the events.

Do these figures suggest that the entire length of the Andaman Island subduction zone ruptured uniformly? Geologically, would you expect the entire length to rupture uniformly?

Exercise: 10.1

Other large Sumatran earthquakes with information on Wikipedia include the: 1833 Sumatra earthquake; 1861 Sumatra earthquake; 1797 Sumatra earthquake; 2005 Nias Earthquake; March 2007 Sumatra earthquakes; September 2007 Sumatra earthquakes; August 2009 Sumatra earthquake; and May 2010 Northern Sumatra earthquake.

```
# Load libraries and Phuket catalogue
library(PtProcess)
library(mapdata)
data(Phuket)

# Scale point size by magnitudes
pointSize <- Phuket$magnitude - 4.95

# Make blank plot
usr <- c(min(Phuket$time), max(Phuket$time), min(Phuket$latitude),
max(Phuket$latitude))
mai <- c(0.85, 0.85, 0.05, 0.15)
plot.new()
par(mai=mai, usr=usr)
title(xlab="Time [days]", ylab="Latitude")
axis(1)
axis(2)
box()

# Define Big event IDs and times
big$id <- c(35, 511, 733, 972, 1124)
big$time <- c(360.040, 452.673, 570.654, 1350.465, 1516.358)

# Loop over all events
for (j in 1:length(Phuket$latitude)) {
    # Loop over all Big events to colour 30 day windows differently
    col <- 1
    for(k in 1:5) {
        if((Phuket$time[j] > big$time[k]) & (Phuket$time[j] < big$time[k]+30))
            col <- k+1
    }
    # Plot points
    points(Phuket$time[j], Phuket$latitude[j], cex=pointSize[j],
           col=col, lwd=pointSize[j], pch=1)
}
```

Table 9 Code to generate a latitude-time event plot coloured as 30 day temporal clusters of events after one of 5 large events directly comparable with 11. Events in black fall outwith one of these 30 day windows.

Catalogues are only ever a snapshot of the data at the point the catalogue was created. By typing `summary(Phuket)` we can find the start and end dates of the catalogue. Which of the events above will be contained within the Phuket catalogue?

What were the estimated sizes of these events? Why was it so long between the events recorded in the 1800's and the recent earthquakes?

Exercise: 10.2 Use the internet to find the major towns along the island of Sumatra that are most exposed to large earthquake events? What is the population of these towns? Are these same towns exposed to tsunamis hazard?

```
library(ssEDA)
library(ssNZ)

a <- subsetcircle(NZ, minday=julian(1,1,1966), maxday=julian(1,1,2006),
minmag=1.0,
                  maxradius=150, centrelat=-37.65, centrelong=179.49)

timeplot(a)
```

Table 10 Code to make histogram of the number of events per year using an SSLib function (See Fig. 13).

Exercise: 10.3 What trends would you expect to see in event depth? Make a cross-section through the data to test this hypothesis (See Sections 7 and 8).

11 Event counts in time

Returning to the New Zealand examples, we now look at how the number of events varies in time. The histograms of the number of events per month are plotted in Fig. 14 for an incomplete catalogue on the left hand side and a complete catalogue on the right. The comparison indicates that the fluctuation in the annual event count arises as a combination of catalogue artefacts and real fluctuations.

In the incomplete catalogue, the number of events per month tends to increase with time reflecting the increase in seismic network coverage with time and its effect on recording smaller magnitude events more faithfully. The complete catalogue removes this increase in event count artefact and now has more consistent numbers of events in each bin apart from two large spikes in recent times, that were also present in the incomplete catalogue.

Question: Determine the time of the largest event. Does the timing of this event correspond to the highest event rates? Are the high event rates that remain in the complete catalogue likely to be real or artefacts?

The indices of the event with the largest magnitude can be found using the following statement

```
n <- which(EastCape1$magnitude==max(EastCape1$magnitude))
```

and the time can the be displayed by typing `EastCape1$time[n]`.

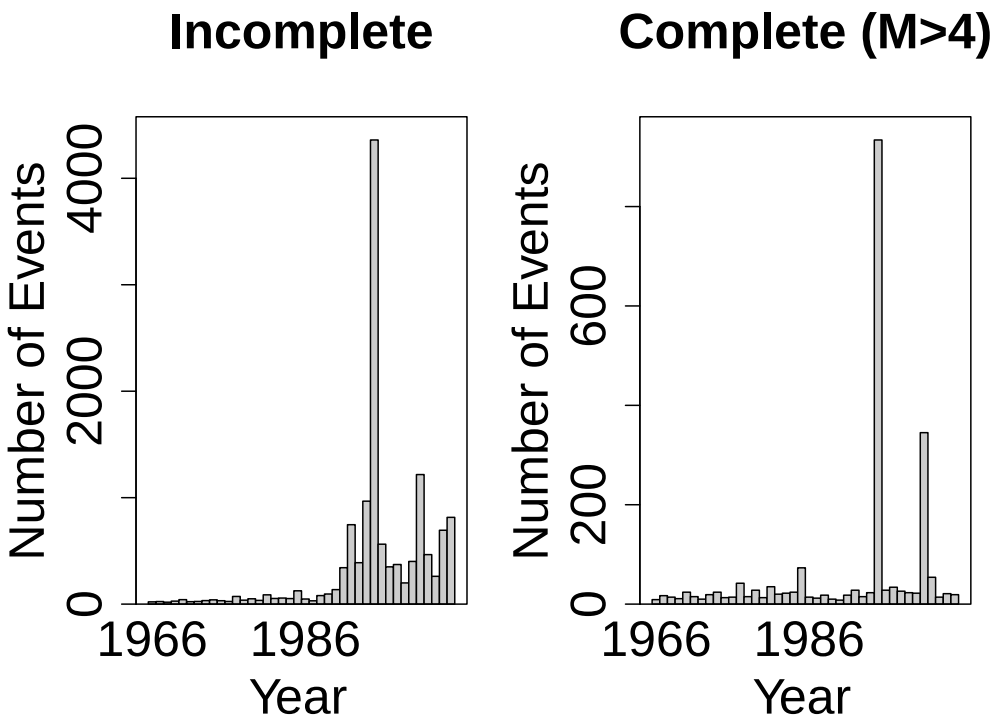

Fig. 13 Annual event count for the East Cape region of New Zealand.

12 Magnitude-time scatter plot

Here we explore the trends in catalogue completeness in more detail using the magnitude-time scatter plot.

The incomplete catalogue (Fig. 14a) shows that the number of small events recorded in the catalogue is increasing with time reflecting improvements in the design and increasing coverage of seismometers. The trends may be gradual, but more often show sudden improvements at lower magnitudes (as occured in Fig. 14a) due to specific projects to increase the recording networks capability. The correlation with the increasing number of events per year in the last section is clear (Fig. 13a).

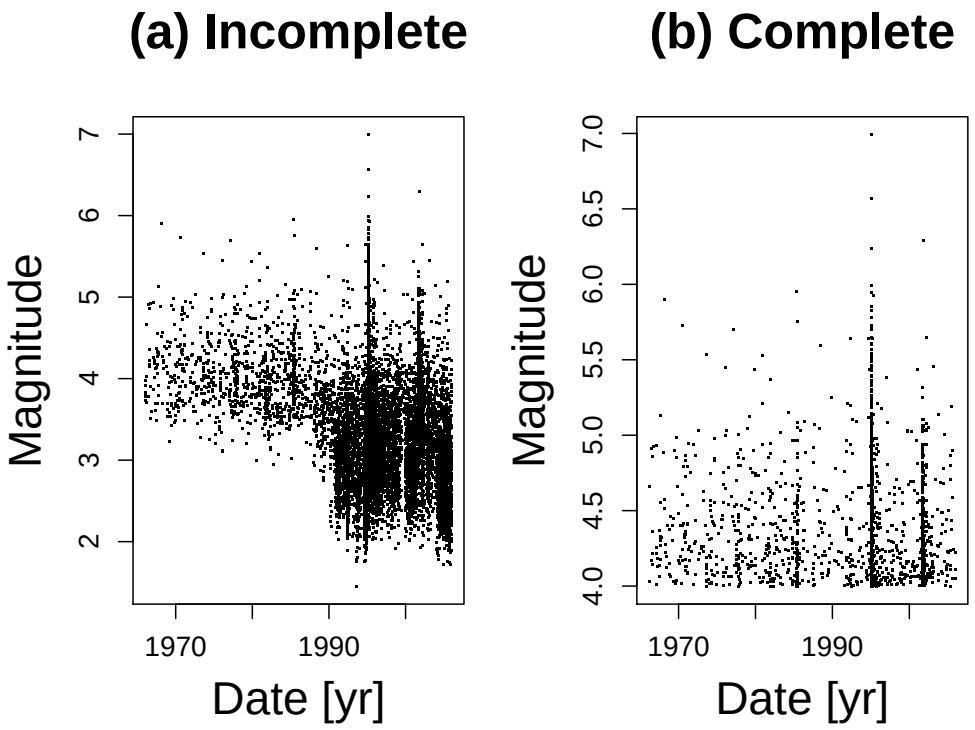

Fig. 14 Scatter plot of event magnitudes with time. By default, times are measured relative to 1 January 1970.

```
library(ssEDA)
library(ssNZ)

a <- subsetcircle(NZ, minday=julian(1,1,1966), maxday=julian(1,1,2006),
minmag=1.0,
                  maxradius=150, centrelat=-37.65, centrelong=179.49)
as.catalogue(a1, catname="EastCape1")

plot(EastCape1$time,EastCape1$magnitude, pch=".", cex=2.5, main="Incomplete",
     xlab="Time since 1 January 1970 [days]", ylab="Magnitude")
```

Table 11 Code to make a scatter plot of the recorded magnitude of events as a function of time. (See Fig. 14).

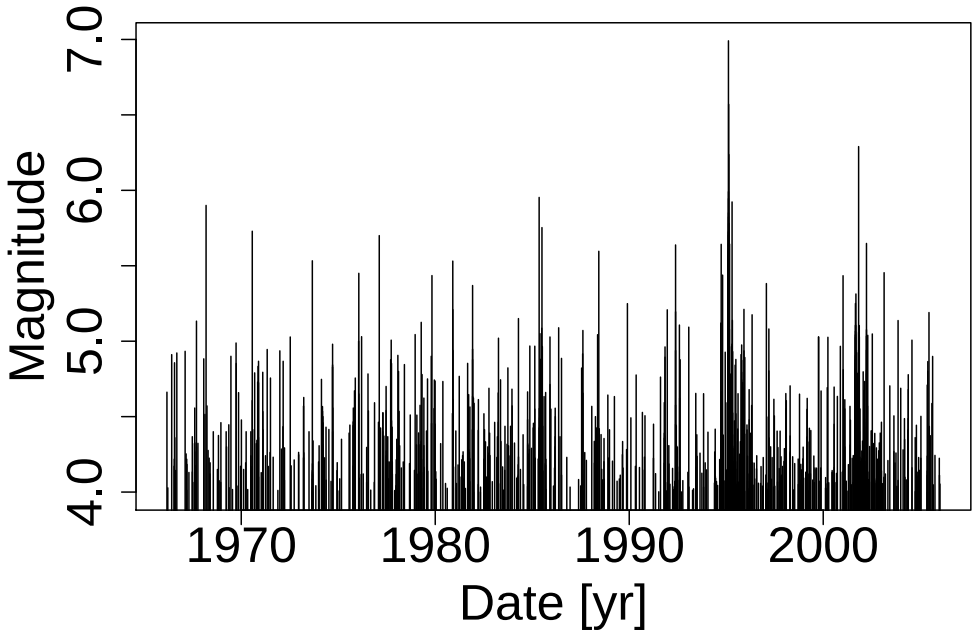

Fig. 15 Comb plot of event magnitudes with time for the same complete catalogue as in Fig. 14b. See Table 12 for the corresponding R code.

13 Comb plot

Comb plots represent individual earthquakes as a vertical bar positioned at the time the event occurs and with a height corresponding to the magnitude of the event. They give an indication of how clustered events are in time and which periods were experiencing extreme events. Intense activity can often be seen after large magnitude events due to triggered events. The comb plot shows the time series is a "point process" of discrete events of different size occuring at irregular time intervals.

```
library(ssEDA)
library(ssNZ)

a <- subsetcircle(NZ, minday=julian(1,1,1966), maxday=julian(1,1,2006),
                  minmag=4.0, maxradius=150, centrelat=-37.65,
centrelong=179.49)
as.catalogue(a, catname="EastCape1")

plot(EastCape1$time, EastCape1$magnitude, type="h", cex=2.5, ylab="Magnitude")

#   Alternatively
magnitude.time(a)
```

Table 12 Code to make a scatter plot of the recorded magnitude of events as a function of time. (See bottom row of Fig. 15).

14 Histogram of interevent times

Using the time-series at which events occur, we can calculate the interevent times for successive events. A variety of histograms of these interevent times are plotted in Fig. 16, all constructed from the same dataset using incremental data. Note how simple choices of linear or logarithmic binning on the x-axis, and linear or logarithmic scaling in the count data on the y-axis, result in radically different representations of the the same data. This is essentially a filtering of the data which emphasises certain features of the initial data whilst downplays other features. A key part of an exploratory data analysis is to push and pull the data in this way to critically understand its structure.

There are many ways to present this type of data, some more informative than others, we will take you through the alternatives presented here. The key feature of the interevent time data that makes it difficult to interrogate is that the duration of the interevent times vary over several orders of magnitude.

The first graph (Fig. 16a) is the simplest. We bin the raw IET data into uniform linear bins and plot the frequencies in linear space also. We see a large spike in the first bin that decays away rapidly. The problem with this version of the histogram is that it only resolved the longest IETs satisfactory, all of its which are 2 orders of magnitude smaller than the largest IETs are lumped into that first bin. We do not really see the structure of the data.

We can improve upon this slightly by plotting the frequencies on a log scale (Fig. 16b) which allows us to see the form of the decay, which is roughly exponential (straightish line on a log-linear graph), but this still does not allow us to see the structure in the smaller IETs.

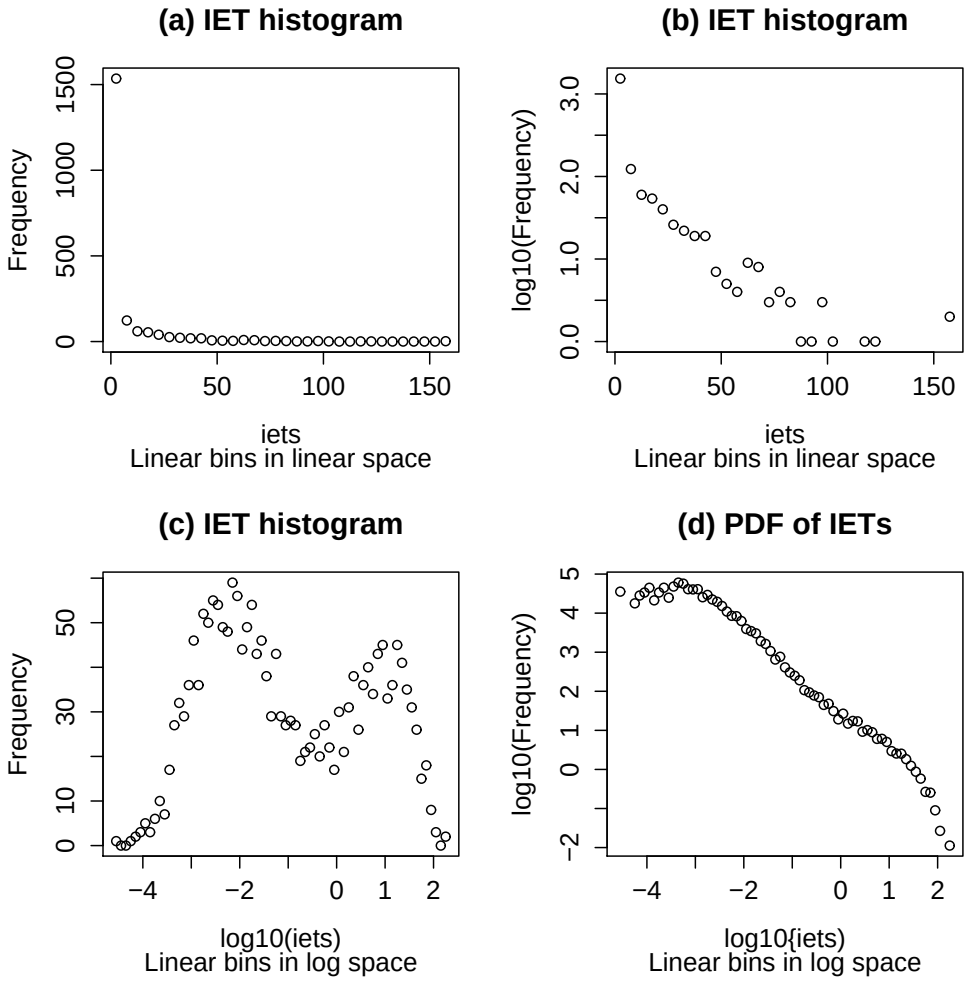

Fig. 16 Histogram of interevent times (a) linear bin widths, linear frequency (b) linear bin widths, log frequency (c) log bin widths, log frequency and (d) a PDF which is log bin widths with the probability given by the log frequencies each normalised by bin width and a global normalisation which ensures that the probability sums to 1. See Table 16 for the corresponding R code.

Rather than binning the IETs in linear bins, we can take the log of the data and then bin them in linear bins. This has the advantage that in linear space the bins have a non-uniform width; in particular the smaller the IETs, the smaller the bins. So, rather than all of the smaller IETs being lumped into one bin, they are spread out over ever smaller bins. We can now clearly see the structure of the interevent times across the entire range (Fig. 16c). By transforming the data prior to binning we respect that the data is spread over many orders of magnitude and introduce a desirable feature to the histogram which is that all of the bins have a similar level of

```
# Load SSLib libraries
library(ssEDA)
library(ssNZ)

# Extract catalogue subset
a1 <- subsetcircle(NZ, minday=julian(1,1,1976), maxday=julian(1,1,2006),
minmag=4.0,
                    maxradius=150, centrelat=-37.65, centrelong=179.49)
as.catalogue(a1, catname="EastCape")

# Extract event times and make interevent time list
times <- EastCape$time
times1 <- as.numeric(times[1:length(times)-1])
times2 <- as.numeric(times[2:length(times)])
iets <- times2-times1

# Plot histogram with 100 bins
hist(log10(iets), n=100)
```

Table 13 Code to first plot a histogram of the interevent time between successive events (See Fig. 16).

occupancy so that the structure is easy to understand. This is the authors' preferred way to plot this data.

A fourth way to plot the data is to create a probability density function (PDF). The area in each bin represents the probability of occupation of that bin; since the bin widths vary, we need to normalise by bin width for the areas to be proportional to the probability of occupation. Further, since the PDF represents the probability of occupation, the area under the curve must sum to 1. All of the previous plots were histograms which use the raw integer counts.

Notice that in the code snippet (Table 13) the log of the data was taken prior to creating the histogram; this was done because the IETs span several orders of magnitude. It is easier to interpret the distributions in this log space because the bins have similar counts and there are few empty bins within the distribution. The bins have a uniform with in log space which means they have log distributed widths in linear space (i.e. the bins for short interevent times are narrower in linear space than those for long interevent times).

Exercise: 14 Starting with the code snippet in Table 13, try to reproduce each of the histograms in Fig. 16. Describe what each of the histograms is describing in the data; what features of the binning and plotting draw out the features and what features of the data are not well described by each histogram.

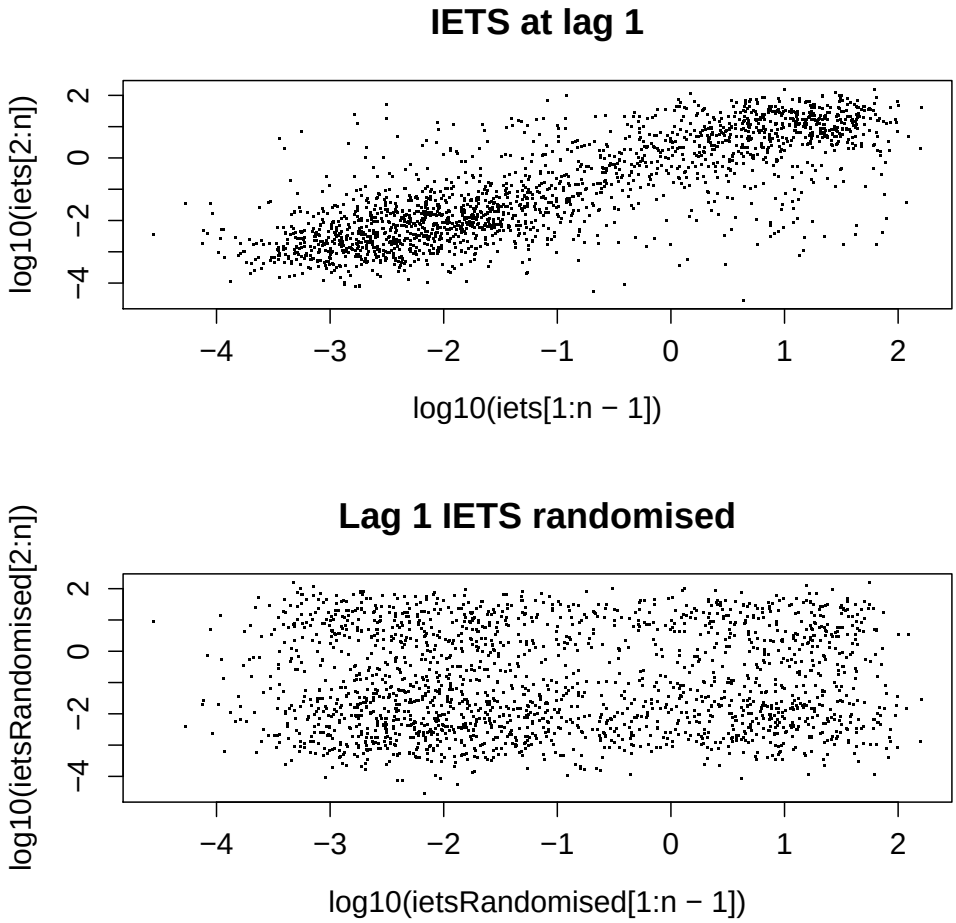

Fig. 17 Demonstration of the correlation between successive interevent times. (a) Scatter plot of successive interevent time pairs shows a linear trend and (b) Scatter plot of randomised interevent time pairs demonstrates the loss of the linear trend in (a) supporting that the correlation is real. See Table 14 for corresponding R code.

14.1 Lagged interevent times

It is interesting to ask the question whether the whether an interevent time is correlated with the previous interevent time. We can graphically investigate this by plotting $(t_i - t_{i-1})$ vs $(t_{i-1} - t_{i-2})$ (Fig. 17).

We can see a linear trend running through the lagged interevent times. We can start to test whether there is really a linear trend in this data by randomly shuffling one of the sequences of interevent times and replotting the graph. We can see that

```
# Load SSLib libraries
library(ssEDA)
library(ssNZ)

# Extract catalogue subset
a1 <- subsetcircle(NZ, minday=julian(1,1,1976), maxday=julian(1,1,2006),
                   minmag=4.0, maxradius=150, centrelat=-37.65,
centrelong=179.49)
as.catalogue(a1, catname="EastCape")

# Extract event times and make interevent time list
times <- EastCape$time
times1 <- as.numeric(times[1:length(times)-1])
times2 <- as.numeric(times[2:length(times)])
iets <- times2-times1

# Randomise interevent time list
ietsRandomised <- sample(iets, replace=FALSE)

# Plot successive interevent times
par(mfrow=c(2,1))
n <- length(iets)
plot(log10(iets[1:n-1]),log10(iets[2:n]), pch=".", cex=2, main="IETS at lag 1")
plot(log10(ietsRandomised[1:n-1]), log10(ietsRandomised[2:n]),
     pch=".", cex=2, main="Lag 1 IETS randomised")
```

Table 14 Code to first plot the interevent time of successive events then the interevent time of randomised event pairs (See Fig. 17).

the linear trend has now been lost. This idea of randomising data which shows a trend is the starting point for calculating bootstrapped

confidence intervals, where confidence intervals are generated statistically by random sampling.

This structure in interevent times indicates a history dependence in earthquake catalogues.

15 Summary

To use the Statistical Seismology approach to learn about the properties of earthquake catalogues and interpret them as physical phenomena you require a combination of statistical experience, computational skill, awareness of bias and ????.

In this tutorial you should have learnt how to:

– Manipulate earthquake catalogues;

– Explore the properties of the catalogue by plotting maps, cross-sections, histograms and scatter plots;
– Calculate the b-value from an earthquake catalogue when the magnitude completion threshold is known;
– Identify some examples of bias that exist in earthquake catalogues; and
– Use R and SSLib to perform these analyses.

This article will have shown you different ways you can plot and interrogate a new earthquake catalogue. When approaching a new problem, you do not have to run all of these analyses, rather you will have to choose the most appropriate tools for investigating the features you are interested in.

Now you need to practice these skills using different earthquake catalogues.

16 Exercises

16.1 Large New Zealand earthquakes

The location, magnitude and timing of large historic earthquakes in New Zealand can be found on the New Zealand Geonet website. Take catalogue subsets around some of these events and apply some of the methods presented in this article to illustrates spatial and temporal trends.

16.1.1 EDA on different tectonic settings

In this tutorial we want you to explore the use the EDA tools outlined above on different tectonic settings.

A Appendix: Loading packages required for running the code

The code snippets included in the text should run directly within R provided you have R, SSLib and the associated libraries installed. Libraries starting with the "ss" prefix are available from the SSLib website. Other packages that are required for specific routines are MASS (Venables and Ripley 2002). In order to use the 3D visualisation interface for viewing events that is provided by SSLib, we also require ggobi and the R package to interface to rggobi (Lang *et al.* 2010).

Information on how to install packages can be found at the R Installation and Administration page under Add-on packages. The sslib packages need to be installed by down loading from the homepage, not through the CRAN online repository.

B Appendix: Loading different catalogue data into SSLib and R

Several earthquake catalogues are provided on the SSLib website that are ready for immediate. These catalogues cover Southern California (ssSCEC), New Zealand (ssNZ), Italy (ssItaly) and the global PDE (Preliminary Determinations of Epicentres) catalogue (ssPDE). These can be loaded into an R session by typing, for example

```
library(ssNZ)
```

directly on the R command line.

However, we don't want to restrict our analysis to only the data available from SSLib so we need to know how to import a new catalogue from a text file. Importing a new set of data into an SSLib catalogue can be tricky, primarily because the raw data are held in a variety of non-standard formats.

The subset simply records the parameters used to make the subset (e.g. spatial-temporal-magnitude boundaries), and the index is events within the subset.

B.1 Phuket

As well as the standard catalogues in SSLib, there is also a Sumatran catalogue centred on Phuket which is located within the ptprocess library. This can be simply loaded using:

```
library(PtProcess)
data(Phuket)
```

The types of data held within this catalogue can be seen using

```
summary(Phuket)
```

These data can then be accessed directly using, for example,

```
Phuket$magnitude
Phuket$time
Phuket$latitude
```

B.2 Mount St. Helens

The The Pacific North West Seismic Network (PNSN) operates seismograph stations and locates earthquakes in Washington and Oregon. Their web site provides information on Pacific Northwest earthquake activity and hazards including the provision of a Mount St.Helens catalogue.

We stored the Mount St Helens earthquake data to a file named "EQoriginaldata.helens.txt". This is exactly how the data are in their webpage.

We have made some changes to this file in order to import it into R. The file that contains the changes is named "EQdata.helens.txt". Note the differences. We are using the scan function to read the data from the text file ("EQdata.helens.txt"). The file in R will be of mode list.

The list should contain the variables year, month, day, hour, minute, and second. These will be replaced in the resulting catalogue with one variable called time, being the number of days (and fractions) from some origin. Then we use the function as.catalogue to create the catalogue named in this example "M.Helens".

The first element in the scan function is the path of the file that contains the data. Note that the `sep` argument in the `scan` function in this example is equal to `","`, i.e. it is a CSV file. One could also use the blank character as the break between fields if there are no spaces within any of the fields.

B.3 Choosing subset regions

Frequently, one wants to analyse fairly small parts of earthquake catalogues. There are four functions provided to subset catalogues: `subsetcircle`, `subsetpolygon`, `subsetsphere` and `subsetrect`. More information on each of these can be found by typing, for example,

```
?subsetcircle
```

```
library(ssEDA)

# The first element in the scan function is the path of the file that contains
the data.
y <- scan("/dir/path/to/file/EQdata.helens.txt", sep=",",
          what=list(year=0, month=0, day=0, hour=0, minute=0, second=0,
latitude=0,
                    longitude=0, depth=0, magnitude=0, comments=""))
as.catalogue(y, catname="M.Helens", dp.second=1)
```

Table 15 Creates a catalogue containing the Mt. St. Helens dataset.

on the R command line. An alternative is to type `help.start()` on the command line. This will open the web browser an give access to the entire R documentation. You can search for various functions or descriptions, or look at all of the available functions within each installed package on your system.

Google Earth is a useful tool to help determine the required latitude and longitude boundaries of a region when subsetting a catalogue. This tool can be used in several ways.

- Turning on the Grid option in the View dropdown menu adds labelled contours of latitude and longitude which fine as you zoom in.
- The latitude and longitude of the the cursor point are recorded in the bottom left of the viewing pane.
- When you drop a pin, the latitude and longitude of that pin can be obtained from it's properties dialogue.
- The ruler tool can be used to calculate a useful region length scale.

Note that by default, Google Earth works in degrees,minutes,seconds whereas SSLib works in decimal degrees. The Google Earth coordinate scheme can be set to decimal degrees in the Options dialogue (accessed from the Tools dropdown menu) which allows you to use the values directly.

C Appendix: Plotting in R

Excellent examples of how to plot different types of graph in R can be found in the R Graph Gallery which contains sample code snippets.

D Appendix: Examples of Applications in the Literature

Use of confidence intervals on frequency magnitude plots to test the hypothesis that there is evidence for characteristic earthquakes, see Naylor *et al.* (2009) and Harte (2010).

References

Aki K (1965). "Maximum Likelihood Estimate of b in the Formula $\log N = a - bM$ and its confidence limits." *Bulletin of the Earthquake Research Institute*, **43**, 237–239. 9

Crawley MJ (2007). *The R Book*. Wiley, Chichester. ISBN 978-0470510247. 4

Gulia L, Wiemer S, Wyss M (2010). "Catalog artifacts and quality control." *Community Online Resource for Statistical Seismicity Analysis*. URL http://dx.doi.org/10.5078/corssa-93722864. 6

Harte D (2010). "PtProcess: An R package for modelling marked point processes indexed by time." *Journal of Statistical Software*, **35**, 1–32. URL http://www.jstatsoft.org/v35/i08/. 41

Harte D, Brownrigg R (2010). *ssBase: Base Functions for SSLib*. Statistics Research Associates, Wellington. R package version 2.2-4, URL http://www.statsresearch.co.nz/software.html. 3

Lang DT, Swayne D, Wickham H, Lawrence M (2010). *rggobi: Interface between R and GGobi*. R package version 2.1.16, URL http://CRAN.R-project.org/package=rggobi. 39

Mignan A, Woessner J, Schorlemmer D (2010). "Completeness magnitude in earthquake catalogs." *Community Online Resource for Statistical Seismicity Analysis*. URL http://dx.doi.org/10.5078/corssa-00180805. 11

Naylor M, Greenhough J, McCloskey J, Bell A, Main I (2009). "Statistical evaluation of characteristic earthquakes in the frequency-magnitude distributions of Sumatra and other subduction zone regions." *Geophysical Research Letters*, **36**, L20303. doi:10.1029/2009GL040460. 41

Team RDC (2010). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org. 3

Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition. ISBN 0-387-95457-0, URL http://www.stats.ox.ac.uk/pub/MASS4. 39

Woessner J, Hardebeck JL, Haukkson E (2010). "What is an instrumental seismicity catalog?" *Community Online Resource for Statistical Seismicity Analysis*. URL http://dx.doi.org/10.5078/corssa-38784307. 6