Practical 2

R version

August 8, 2017

Introduction

In this practical you will explore postural dynamics and specific acceleration in the time and frequency domains. You will use spectrograms to find intervals of movement and will use filters to separate these into distinct frequency bands. Finally you will quantify several aspects of movement including ODBA and dominant stroke/stride frequency.

Learning objectives:

i. Become familiar with plotting and interpreting movement data in the frequency domain.
ii. Use spectrograms to locate intervals with periodic movement.
iii. Use spectral averages to characterise movements.
iv. Continue developing techniques for checking data quality and ameliorating problems.
v. Learn how filters can be used to emphasise different features of movement data.
vi. Learn how to characterise propulsion movements using ODBA and MSA.
vii. Gain an understanding of how filtering and appropriate sampling rates can help in detecting transient events such as prey capture attempts.

Section 1. Analysing accelerometer and magnetometer data in the frequency domain

1.1 Quality checks:

For this practical we will use data from a collar-mounted tag on a sheep. We don't know much about these data so will need to check them for quality and then infer what we can about the deployment and the animal's behaviour.

First load in the data set and check what variables there are and their sampling rate.

```r
#(modify path to match your computer)
setwd('C:/Users/Stacy DeRuiter/Dropbox/TagTools/data')
sheep <- load_nc('oa14_319a_raw')
```

A brief R note: Avoid copy/pasting code with special characters like _ In some cases they don't translate correctly and you will get a "no such file" error.

Before exploring the accelerometer and magnetometer data, we need to do a quality check. Use check_AM to get the two field strengths and the inclination angle. Plot these and check them against what you expect for the general location where the data were collected (hint: it is in the info structure). Are A and M trustworthy?

```r
AMcheck <- check_AM(A=sheep$A,M=sheep$M)
```

The magnetometer data needs an offset correction:

```r
Mnew <- fix_offset_3d(sheep$M)
#need to get Mc and Gm from this and store in sheep
```

```
sheep$Mc <- Mnew$X
sheep$Gm <- Mnew$G
```

```
AMcheck <- check_AM(sheep$A,sheep$Mc)
```

Re-run the quality check on Mc. Has the field strength or inclination improved?

If the field strength is ok for A and M but the inclination angle is wrong, this is a strong indication that the magnetometer and accelerometer axes don't match. Assuming that the accelerometer axes are correct (Forward-Right-Up), I found by trial and error that the magnetometer axes mapped as follows:

```
Sensor axis |   Tag axis
------------------------
x                  |  -x
y                  |   z
z                  |   y
```

These can be corrected by adding a map matrix to the calibration structure Gm as follows:

```
sheep$Gm$map = matrix(c(-1, 0, 0,0, 0, 1,0, 1, 0),
                      nrow=3, ncol=3, byrow=TRUE)
```

To re-do the calibration to use this map, use apply_cal. This applies any calibration fixes in Gm to the magnetometer. In this case, we have an offset fix and a map fix.

```
sheep$Mc <- apply_cal(sheep$M,sheep$Gm)
```

Now check the field strength and inclination of A and Mc. Are we ready to go? Bear in mind that the inferred inclination angle from accelerometer and magnetometer data usually has a lot of variability.

This calibration process should now be documented in the metadata in Mc. Display the metadata (just by typing Mc and enter in Matlab/Octave) to verify that it has collected the correction steps that you added.

1.2 Plotting and interpreting the spectrogram

Now A and M are trustworthy, use plott to plot them both. What can you infer about the activity level of the animal as a function of time? When does it seem to be resting and active?

Pick the axis of A that seems to be the most variable when the animal is active and plot the spectrogram of this axis. For example, if you want to use the x axis:

```
make_specgram(sheep$A$data[,1], nfft=1024,
              sheep$A$sampling_rate)
```

This will plot a spectrogram with 1024 length FFTs. What is the frequency and time resolution of this spectrogram? Hint: one is 1024 divided by the sampling rate and the other is the inverse of this.

Zoom in to the active part of the data and try to find an interval with periodic movement (e.g., walking or running). What should this look like in the spectrogram? Hint: think about the Fourier transform of a periodic signal.

Find time cues (in seconds) for the start and end of the interval with periodic movement. In R, you can use `locator` to click on the start and end point on the spectrogram:

```
g <- locator(n=2, type='o')$x
```

Note the dollar-sign-x at the end of the line selects just the x positions of the points clicked (and discards the y positions). If you just read the start and end times off the spectrogram without using ginput, make a variable g (replacing start and end with times in seconds):

```
g <- c(start,end)
```

Use these times to crop out the interval into new variables Aseg and Mseg:

```
Aseg <- crop_to(sheep$A,tcues=g)$X
Mseg <- crop_to(sheep$Mc, tcues=g)$X
```

Note: be sure to use your corrected magnetometer data, Mc, not the original data. Use plott to check that you cropped the right section.

1.3 Calculating spectral averages

The dominant stroke/stride frequency is a useful measure of how rapidly an animal is moving. This is also useful for designing filters such as needed to calculate ODBA. The toolbox function dsf calculates the dominant stroke/stride frequency on a data segment. Try it with both Aseg and Mseg:

```
Adsf <- dsf(Aseg)
Mdsf <- dsf(Mseg)
```

How do the results compare - are the estimated stride frequencies similar? The magnetometer is not sensitive to specific acceleration so how can the stride frequency be estimated from the magnetometer? Go back to your plott of Aseg and Mseg to verify that there is a stride-rate signal in Mseg. Which axis is it in? Any idea what kind of movement it might be picking up?

To get more detail on the spectrum of this movement, we can compute a spectral average, i.e., the average frequency content over the entire segment. The tagtool `spectrum_level` does this on all three sensor axes:

```
fs <- Aseg$sampling_rate
slA <- spectrum_level(Aseg$data, nfft=512, sampling_rate=fs)
slM <- spectrum_level(Mseg$data, nfft=512, sampling_rate=fs)
```

This time we are doing 512 length FFTs. `spectrum_level` divides the data up into as many blocks of 512 samples as it can, computes the FFT of each of these, and averages the power of the results. It returns the spectral average in decibels i.e., 20*log10(signal size), which is good for sound which has a high dynamic range but is not always so useful for movement data. We can reverse the decibels by doing $10^{Sa/20}$

In R:

```
10^(Sa/20)
```

The units of the spectrum will then be in $m/s^2/\sqrt{Hz}$ for acceleration and $\mu T/\sqrt{Hz}$ for magnetometers.

You can plot the spectral averages for Aseg and Mseg on the same figure:

```
#make simple plot with 2 panels in two rows
par(mfrow=c(1,2))
#some relatively nice colors, color-blind OK
line_colors <- c("#000000", "#009E73", "#9ad0f3",
                 "#0072B2", "#e79f00", "#D55E00")
plot(slM$f, 10^(slA$SL[,1]/20), type='l')
for (c in 2:ncol(slA$SL)){
  lines(slM$f, 10^(slA$SL[,c]/20), col=line_colors[c])
}
plot(slM$f, 10^(slM$SL[,1]/20), type='l')
for (c in 2:ncol(slA$SL)){
  lines(slM$f, 10^(slA$SL[,c]/20), col=line_colors[c])
}
```

Just to show you know what you’re doing here, add a label to the horizontal axis saying what it is and what is the unit (in R, add input `xlab=’My New Axis Label’` to `plot`.

How do the spectra of Aseg and Mseg compare? (note: ignore any frequencies below 1 Hz - these are from slow changes in orientation). Looking first at the axis of each sensor with the strongest fundamental signal: is it the same axis for both sensors? If so, why? Do you see a 2nd harmonic in either sensor? If not both, why do you think this might be? What spectral components (if any) are in the other axes?

1.4 Inferring type of activity from time and frequency data

Re-plot the spectrogram of the full A signal and zoom in to the active time to find an interval that has non periodic movement. What should that look like in the spectrogram?

Find time cues for the interval (e.g. using `locator`) and crop A and M as before. Call the results A2 and M2 and plot them to check you got the right interval. What can you tell about this movement from the plots?

Bearing in mind that we don’t know how the tag axes map to the body axes, compare the mean levels of A2 and Aseg (by eye should be enough). Is there a consistent difference in orientation between the two activities? Even without knowing how the tag is oriented on the body, which Euler angle do you think is changing? Hint: think about the rotational degrees of freedom of a neck with a collar on it.

Calculate the spectrum levels for A2 and M2 and plot them. How do they compare with each other and with the spectra for walking/running?

The big question: In this mystery activity, is the animal (a) grazing, (b) searching or (c) sitting and chewing? What evidence do you have to support your inference?

Section 2. Using filters to analyse, quantify and detect events in accelerometer data in the time domain

2.1 Complementary filtering:

For this practical we will use data from a suction cup tag attached to the back of a beaked whale. The data is in `testset1`. Use `load_nc` to read it in and `plott` to inspect it. As always, do a quality check on A and M. Also check that P is correctly calibrated (just by zooming in on the data plot) so that the data says the animal is at the surface when it is at the surface.

This dataset includes a shallow and a deep dive. We want to infer the function of these by looking for locomotory effort and sudden changes in acceleration that could be indicative of prey capture attempts. We are also going to look for any changes in swimming gait throughout the dataset.

To separate slow orientation changes from postural dynamics in locomotion, we need to choose a filter cut-off frequency below the stroke frequency. We can estimate the dominant stroke frequency using tagtool: `dsf`. There is a bout of steady swimming between minutes 35 and 42 in the data. Use `crop_to` to pick out the accelerometer data in that interval and run `dsf` on it. Plot the segment of acceleration data to make sure you got the right one. What is the stroking rate?

```
Ac <- crop_to(md$A, tcues = c(35*60, 42*60))
```

A good starting choice for the filter cut-off frequency is 70 percent of the stroking rate. Call this value fc. Run a complementary filter on the full A and M data to separate the slow and fast time-scales:

```
fc <- 0.38*0.7
```

```
Af <- comp_filt(md$A,fc=fc)
Mf <- comp_filt(md$M,fc=fc)
```

The complementary filter returns a cell array containing two data matrices: the low-pass filtered data and the high-pass filtered data. Each of these is a three-column matrix because the filter is run on each column of the input data. So it is like you now have two accelerometers and magnetometers in the tag - one pair that was only sensitive to low-frequencies and the other that was only sensitive to high frequencies.

To get each of matrices out of the cell array, do:

```
Alow <- Af[[1]]
Ahigh <- Af[[2]]
Mlow <- Mf[[1]]
Mhigh <- Mf[[2]]
```

The sampling rate of all of these is the same as for the original data. For simplicity, let's make a variable equal to the sampling rate:

```
fs <- md$A$sampling_rate
```

Use plott to plot the two filtered accelerations:

```
plott(X=list(Alow=Alow, Ahigh=Ahigh), fsx=fs)
```

These two versions of acceleration are sometimes called 'static' and 'dynamic' acceleration in the literature although these terms are not very well defined. Looking at the plot, you can see straightaway that the large slow changes in acceleration have been removed from Ahigh in the bottom panel - these slow changes are hopefully mostly to do with orientation so by removing them, we focus on the high frequency specific acceleration and postural dynamics. Zoom in to the top panel to verify that most of the fast changes in acceleration have also been removed from Alow. For example, zoom into the section of steady stroking that you used for dsf - you should only see the stroking in Ahigh, not in Alow.

2.3 Locomotion style:

We want to characterise the locomotion style of the animal. We will do this by comparing the high frequency acceleration and magnetometer data for two intervals in the ascent from the deep dive. To explore these data, first make a plot of depth, and the high frequency acceleration and magnetometer:

```
plott(X=list(Depth=md$P,Ahigh=Ahigh, Mhigh=Mhigh),
      fsx=fs, r=c(TRUE, FALSE, FALSE))
```

Zoom in on the ascent of the deep dive and see if you can identify intervals in which the animal appears to be just steady swimming - not foraging. Do you see any changes in the swimming style through the ascent? In particular, check out the swimming styles in time intervals 36-38 and 56-58 minutes. Crop out the high frequency acceleration data in each of these intervals using:

```
intvl1 <- c(36*60, 38*60)
intvl2 <- c(56*60, 58*60)
Ah1 <- crop_to(Ahigh,sampling_rate=fs,
               tcues=intvl1)$X
Ah2 <- crop_to(Ahigh,sampling_rate=fs,
               tcues=intvl2)$X
```

Notice that the y-axis scales may vary a bit in the plot output. (Ability to have user-specified y-axis limites is under development for the R version of `plott`.)

```
plott(X = list(Ah1=Ah1, Ah2=Ah2),fsx=fs)
```

Comparing these two intervals of swimming, what do you conclude about the swimming styles? Look at the magnitude of the acceleration (the units are in m/s^2). Does it seem that one swimming style is more energetic than the other?

To substantiate that inference, we will plot the minimum specific acceleration (MSA) for the two intervals. This is computed from the original unfiltered data:

```
ms <- msa(md$A)
```

and then we need to crop out the intervals of interest:

```
ms1 <- crop_to(ms,fs,intvl1)$X
ms2 <- crop_to(ms,fs,intvl2)$X
```

and plot them:

```
plott(X=list(ms1=ms1,ms2=ms2), fsx=fs)
```

Does the MSA support your inference about how much acceleration is generated in each of the swimming styles? Can you now conclude whether the difference between the swimming styles is a result of larger amplitude swimming strokes (i.e., a larger orientation dynamic due to body undulations) or a larger specific acceleration, or a bit of both?

Finally go back and plot the full high frequency acceleration data, Ahigh, to see whether there is active swimming in the shallow dive and, if so, which type of swimming gait is used there.

In case you are interested (and have time), try plotting also the high frequency magnetometer data for the same two intervals of swimming (you will need to first crop Mhigh as you did with Ahigh to get the sections of data). Verify that you can see the swimming motion in these data. How does it compare to the same intervals in the accelerometer (Ah1 and Ah2)? Do you see a similar increase in magnitude in the magnetometer data in interval 2 as you did with acceleration. Does this support your inference about whether specific acceleration or fluking amplitude is changing across the swimming styles?

2.4 Acceleration transients, jerk and dynamic acceleration:

We want to look for indications of foraging during the two dives. Sudden changes in acceleration are often associated with prey capture attempts although they can also arise in other activities such as social encounters, grooming, and in attacks by predators. Rapid changes mean high frequencies and so these will show up in the high-pass filtered acceleration, Ahigh, that you made with the complementary filter. Because we don't know what kind of movement is involved in a prey capture attempt, we don't know which axis of acceleration will be most relevant. It is easier in this case, to just compute the magnitude of the three axes - that way a peak in any or all axes will show up. This gives what is sometimes called the dynamic acceleration (although that term is very ambiguous):

```
DA <- norm2(Ahigh)
```

Another way to emphasise rapid changes in acceleration is by differentiating acceleration to produce the jerk. This is effectively high-pass filtering acceleration with a filter that has a roughly constant slope - the higher the frequency, the more it is emphasised. As with DA, we compute the magnitude of jerk over the three axes to get a single vector.

```
J <- njerk(md$A)
```

Plot DA and J to compare them:

```
plott(X=list(DA=DA, Jerk=J),fsx=fs)
```

Zoom in on the obvious peaks around minute 20 to minute 30 and see how clear they look in DA and J. Which signal gives the lowest noise between the peaks, i.e., the best signal-to-noise ratio? Why should one of the two methods, jerk and DA, perform better than the other? What does that suggest about the frequency content in the acceleration transients?

To follow this up, try decimating the acceleration before computing the jerk. For example, to decimate by a factor of 5, you do the following:

```
Ad <- decdc(md$A,5)
```

What is the resulting sampling rate? Compute the jerk of the decimated acceleration and plot it with the jerk of the full bandwidth acceleration:

```
Jd <- njerk(Ad)
plott(X=list(Jerk=J, dJerk=Jd),fsx=c(fs,fs/5))
```

Zoom in to have a look at the 'signal-to-noise' in the jerk transients. How clear are the jerk transients in the decimated signal compared to the original data rate? What does this tell you about the frequency content in the transients?

To find potential prey capture attempts, we need to run a detector on the norm-jerk signal. Although it is easy to see peaks by eye in the data, detectors require several pieces of information to do a good job. In particular, they need to know the threshold above which a peak is really a peak and the blanking time, i.e., the minimum time that must elapse after a detection before another detection can happen. The tagtools include an interactive peak detector that allows you to choose a threshold and see the effect this has on which transients that are detected. If you don't tell it a blanking time, it will try to choose one that makes sense from the data. Follow the instructions it prints out to change the threshold. You need to pick a threshold that excludes most of the jerk transients during the strong locomotion in the ascent but that still detects most of the jerk transients during the bottom part of the deep dive (balancing false detections and missed detections is often not easy in a detector and is a matter of finding a trade-off that works for your application).

```
pks <- detect_peaks(J,fs)
```

This function returns a structure of information about each peak it detects. It also tells you the threshold and blanking time that it ended up with. For each detection, the start and end times are reported (in samples, not in seconds) and the peak height. You can plot the height and time of each detection along with the dive profile in matlab `plott` but not yet in R (sorry). Try doing it on your own with base graphics or `ggplot2`…

Bearing in mind that some of the jerk peaks might come from strong locomotory strokes, is there strong evidence for foraging in the deep dive? In the shallow dive?

So adding your inference from section 2.3 about locomotion in the shallow dive, what can you conclude about the behaviour of the animal in the shallow dive? Is it resting, traveling or foraging?