STATISTICS ON NON-SMOOTH SPACES: PROBLEMS AND PRACTICALS

TOM M. W. NYE

This document gives brief instructions for practical problems relating to the lectures on statistics in non-smooth spaces. The problems vary in difficulty: a star rating is used to indicate easier ($*$) and harder problems ($***$). There are some pen-and-paper problems as well as computer based problems. The final section of these notes describes some programs which are required for some of the problems. It is anticipated that the lecturer will need to give some help with all the problems, but those labelled ($**$) or ($***$) will require substantial discussion. There are more problems here than time allows: they are intended as suggestions for further exploration individually or in groups.

1. Introduction

(1) ($*$) Prove that any metric tree is CAT(0). (Try to do this directly rather than using the Gromov lemma.)

(2) (a) Let $X = \mathbb{R}^2 \setminus Q$ where $Q = \mathbb{R}^2_{>0}$. Show that X is CAT(0).
(b) Let $Y = \mathbb{R}^3 \setminus Q$ where $Q = \mathbb{R}^3_{>0}$. Is Y CAT(0)?

(3) ($*$) Given two points $x, y \in C_{k\pi/2}$ what is the geodesic $\Gamma(x, y)$? Write a function which takes x, y and returns the length of $\Gamma(x, y)$, and another which returns the point a proportion $s \in [0, 1]$ along $\Gamma(x, y)$. The most interesting cases are $k = 3$ and $k = 5$, and these could be handled with separate functions.

(4) ($*$) Perform multidimensional scaling (MDS) using BHV distances between trees in two experimental data sets. MDS is a method for approximating a set of metric distances between objects via a set of points in Euclidean space, usually $\mathbb{R}^2$ or $\mathbb{R}^3$. The data sets are Bayesian posterior samples described at the end of these instructions. Compute the BHV distances between all pairs of trees in each data set, and use R to carry out MDS. What does the visualization reveal?

(5) ($**$) How often do BHV geodesics pass through codimension-k regions in tree-space for $k = 1, 2, \ldots$? In particular, how often do geodesics pass through the origin (cone-point)? The experimental data sets in the previous practical problem can be analysed to address this problem. Alternatively, trees simulated from some particular distribution could be used to generate the end-points of random geodesics. The `ape` package in R provides the `rtree` function which samples tree topologies using a **coalescent** or **Yule** process. Edge lengths can be assigned using a gamma distribution or reflected normal distribution. The distribution of crossings of codimension-k regions will depend on the exact distribution used to generate the random geodesics. Note that if you could derive any theoretical results about the behaviour of random BHV geodesics, this could probably form the basis of a publication.

2. Quantifying variation

(1) (∗) Implement a function for computing the Fréchet mean on $C_{5\pi/2}$. Use the function to investigate stickiness: how often does the Fréchet mean lie at the origin for suitably simulated data sets? Some thought should be given to how best to simulate data. What happens if you try the Sturm algorithm on $C_{3\pi/2}$?
(2) (∗) Write a function to implement an algorithm for computing the Fréchet mean in BHV tree-space. Test it out on the two sample data sets. Is the Fréchet mean fully resolved in each case? Perform an MDS to visualize the position of the mean in relation to the rest of the data.
(3) (∗∗) Reproduce the derivation and plot of the surface of the locus of the Fréchet mean for the example in the article by Nye et al (arXiv:1609.03045). Can you construct any other interesting examples, perhaps across 4 or more orthants?

3. Stochastic processes

(1) (∗∗) Write a program to simulate Brownian motion on $C_{k\pi/2}$ via random walks. Use this to simulate samples and compare the results with the exact solutions given in Section 3 of the paper *Diffusion on some simple stratified spaces* (see the list of references).
(2) (∗∗∗) Implement an MCMC algorithm to sample from the Gaussian distribution on tree-space described in lectures. The proposal mechanism should be the "spherical distribution" also described in lectures, which has been implemented as a java class. Play around with the variance of the proposal distribution to obtain an MCMC chain which mixes well.
(3) (∗∗/∗∗∗) Use simulation to compare the Gaussian distribution on tree-space with the Brownian motion distribution for comparable variances and the same "source" parameter. This can be done analytically on $\mathcal{U}_4$. For $\mathcal{U}_N$ with $N > 4$, simulation can be used: the previous practical problem can be used to simulate from the Gaussian distribution and a java program is given which produces samples from the random walk. Suppose the source parameter is a tree on which all edges are 1. What is the probability that a tree sampled from either distribution lies in the original orthant? How does this change as the variance parameter increases? And how does it vary with N? Is there evidence of a "phase change"?

Software

A collection of programs have been created specifically to help solve the computer based problems. While they are written in Java, they can all be called directly from the command-line, or invoked from your own software e.g. by using the `system` function in R. Java source has been provided, however, so students with some experience of Java might prefer to use the Java code directly. The software is available from

`http://www.mas.ncl.ac.uk/~ntmwn/teaching.html`

To run the software:

(1) Download all the three jar files into a single directory.
(2) At the command line type:

```
java -classpath ".:path_to_jar_files/*" practicals.PROGRAMNAME
```

(On windows, try using a semicolon rather than the colon in the line above if problems arise).

If you call a program with no arguments, a message will be displayed explaining the arguments required and their syntax. `PROGRAMNAME` is one of the following.

ViewGeodesic: This program either takes a filename containing two Newick strings, or two newick strings on the command line as its input. A window showing the geodesic will be displayed.

CalculateDistances: Compute the BHV distance for every pair of trees in a file.

GetTreeOnGeodesic: Get a tree the tree a proportion $s \in [0, 1]$along a geodesic.

GeodesicPartitionSizes: Takes two trees as input. Outputs the number of splits in each set of splits A_i and B_i forming the partitions of splits which correspond to a geodesic.

MatchTopologies: Takes a single tree and a file containing trees as input. Outputs the proportion of trees in the file whose unrooted topology matches the single reference tree.

SampleRandomWalk: Generates a sample of trees from a distribution generated by random walk.

SampleSphericalDistribution: Samples the spherical "uniform direction, chi-squared distance" distribution described in lecture 3.

E-mail address: tom.nye@ncl.ac.uk

School of Mathematics and Statistics, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK