Computational Statistical Experiments:

STAT 218 - 07S2 (C) Student Projects Report UCDMS 2008/5

This document was completed with the fiscal support from external grants to Dominic Lee. It was typeset by Zhu Sha. All the projects were supervised by Raazesh Sainudiin while he coordinated the second-year course called STAT 218 - 07S2 (C) during Semester Two of 2007 (16/07/2007-15/11/2007) at the

Department of Mathematics and Statistics,
University of Canterbury,
Private Bag 4800,
Christchurch, New Zealand.

Contents

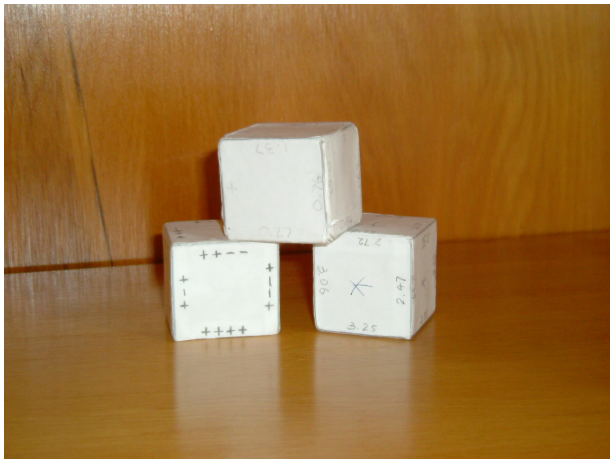
Chapter 1

Investigation of a Statistical Simulation from the 19th Century

Brett Versteegh
and ZHU Sha (Joe)

Abstract

This project is designed to investigate Sir Francis Galton's statistical dice experiment. We constructed Galton's dice according to his prescriptions and tested the null hypothesis that the outcomes from these dice do indeed follow a discrete approximation to the normal distribution with median error one. The inverse distribution function sampler and Chi Squared test are the statistical methodologies employed in this project.



Introduction & Motivation

The report will firstly cover the background and motivation of this project. Secondly, the methodologies used will be explained before outlining the results and subsequent conclusion found by undertaking this experiment. Finally, a potential modification to Galton's method will be examined as a means of sampling from a standard normal distribution.

Background - Francis Galton

Born in 1822, Francis Galton was considered by many, at an early stage, to be a child prodigy. By the age of two, he could read; at five, he already knew some Greek, Latin and long division.

After his cousin, Charles Darwin, published *The Origin of Species* in 1859, Galton became fascinated by it and thus devoted much of his life to exploring and researching aspects of human variation. Galton's studies of heredity lead him to introduce the statistical concepts of regression and correlation. In addition to his statistical research, Galton also pioneered new concepts and ideologies in the fields of meteorology, psychology and genetics.

Background - Statistical Dice

This experiment came about from Galton's need, as a statistician, to draw a series of values at random to suit various statistical purposes. Dice were chosen as he viewed them to be superior to any other randomisation device. Cards and marked balls were too tedious to be continually shuffled or mixed following each draw, especially if the required sample size was large.

The dice he created made use of every edge of each face which allowed for 24 equal possibilities as opposed to the six of a normal die.

For further details on Galton's experiment, please refer to his article "Dice for Statistical Experiments"; *Nature* (1890) No 1070, Vol 42 (This article is available free for download. Please refer to the references section for the website.)

Motivation

The motivation behind this project is to reconstruct Galton's dice using the methods outlined in his 1890 *Nature* article "Dice for Statistical Experiments"and then harness the power of modern computers to determine how effective this technique was for simulating random numbers from the following distribution.

Galton outlines that the samples were taken from a normal distribution with mean zero and median error one. We shall call this distribution Galton's Normal distribution or GN. However, for the experiment

to work, we must use a discrete approximation of the normal distribution, which we will define as Galton's Discrete Normal or GDN. Both will be formally explained in the Methodology section.

To determine the success of this experiment, we formulate the following question as a statistical hypothesis test: "Are our sampled values taken independently and identically from an appropriate discrete distribution which approximates Galton's normal distribution?"

Materials & Methods

Experiment Process

In order to recreate Galton's Dice Experiment, we have chosen to replicate the design he explains in his *Nature* article.

Creating the Dice

We chose to use rimu as it was readily available and inexpensive, unlike the mahogany that Galton had access to. As per his specifications, the wood was cut into six cubes of 1.25 inches (3.2 cm) wide, high and deep, before being covered in a paper template that was designed to fit tightly around the wood. The paper was adhered using general PVA glue.

The only change to Galton's original specification was that we chose to write the values to two decimal places on the faces, as opposed to one decimal place. This was to ensure a higher level of precision when plotting the results.

Collecting the Data

The experiment was carried out by shaking all of the first three dice (dice 1) at once and rolling them across the flat surface of a table top. We interpreted Galton's terminology of the values that "front the eye"to be the results that one can see by looking directly down on top of the dice. The three dice were then lined up into a row and the values called out and entered onto a Notepad document. We used the following formula to calculate the optimal number of trials needed for our investigation:$f(x)_{min} * sample\ size \approx 5$, where $f(x)_{min}$ is the smallest probability for the discrete distribution.

The same rolling process was then performed for dice 2 (two dice at once) and 3 (only one die) with the single exception that we did not need to roll these dice as many times as dice 1.

Statistical Methodology

Firstly, we will define Galton's Normal distribution. As derived from an article published in *Statstical Science*[1], Galton's Normal Distribution has a mean of zero but the variance is not one. Instead, Galton's sample is taken from a half-normal distribution with a "probable error"(median error) of one. This implies that the probability between zero and one is a quarter, allowing us to solve the following equation to determine the variance:

$$\phi(x) = \frac{1}{\sigma\sqrt{2\pi}} exp(-\frac{x^2}{2\sigma^2})$$
$$\frac{1}{4} = \int_0^1 \phi(x)dx$$
$$\frac{1}{4} = \int_0^1 \frac{1}{\sigma\sqrt{2\pi}} exp(-\frac{x^2}{2\sigma^2})dx$$
$$\sigma = 1.4826$$

$\therefore$ GN $\sim$ N (0, 1.48262)

Secondly, we must determine how Galton calculated the values[2] to use on his dice. It was our assumption that he used the midpoints of a set of intervals that partition [0, 1] and we undertook the following processes to confirm this.

We divided the interval [0.5 1] equally into 24, with the last 3 intervals further divided into 24 subintervals. In total, this gave us 21 + 24 intervals to allocate along the y-axis. The midpoint of each interval was taken in order to compute its corresponding x value under the inverse CDF map.

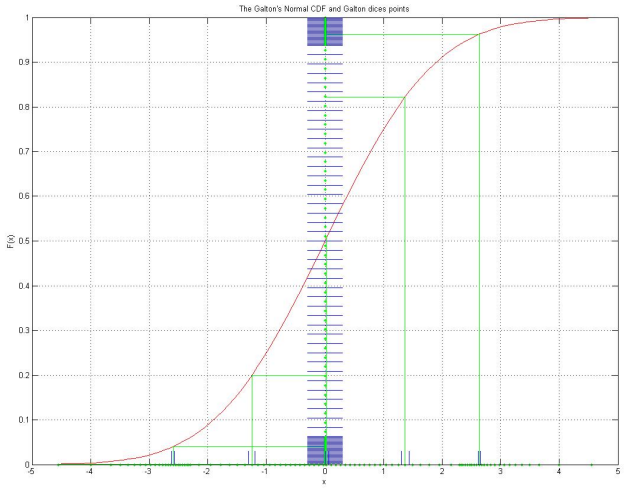


Figure 1.1: Plot showing the midpoints mapping back to specific values on the x axis.

The easiest way to do this would have been to evaluate the inverse CDF function at the midpoints.

[1]Stochastic Simulation in the Nineteenth Century. *Statistical Science* (1991) Vol 6, No 1, pg 94.
[2]See Appendix B.

However, a closed form expression for the inverse CDF does not exist for a Normal distribution. Thus, we applied numerical methods to solve for x (Newton's method).

We believe the midpoint assumption was correct, as the mapped values are very close to Galton's actual figures and the differences can be attributed to an imprecise value for the standard deviation.

Thirdly, we can now determine Galton's discrete approximation to the Normal. This is necessary as the values drawn from throwing Galton's dice come from a discrete distribution, not the continuous Galton Normal. In doing this, we are also able to define our null hypothesis formally: $H_0 : x_1, x_2, \ldots, x_n IID \sim GDN$ Galton's Discrete Normal (GDN) is an approximation to Galton Normal (GN).

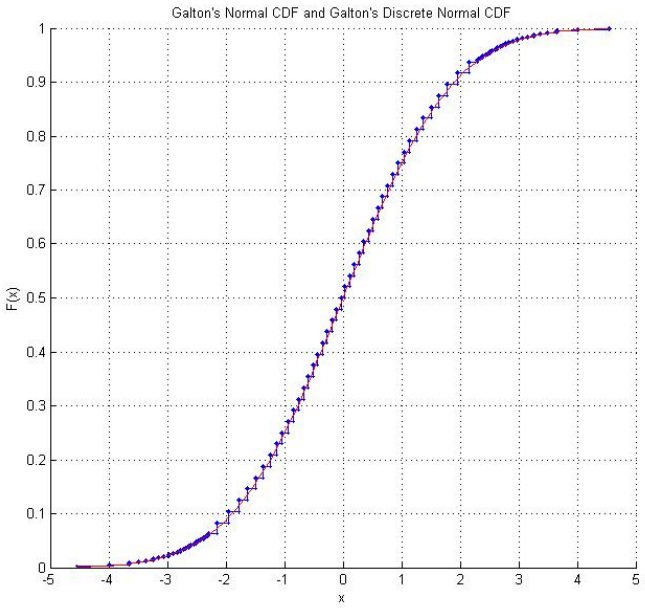


Figure 1.2: Plot showing both the GN and GDN CDFs. They are very similar.

Fourthly, as the distribution is now discrete, we can apply the Chi Squared Test to evaluate our null hypothesis. The test used had the following parameters: Degrees of Freedom: $90 - 1 = 89$ $\alpha = 0.05$; Critical Value = 112.

Results

Once the experiment was complete and the results collated, they were run through a methodological tester to ensure all values were correct. Testing the data involved running all our sampled values through a `Matlab` function which checked each number against Galton's 45 possible values. Any values that did not match were outputted as ones and the erroneous data were removed before a graph was plotted to measure how well our experiment sampled from GDN.

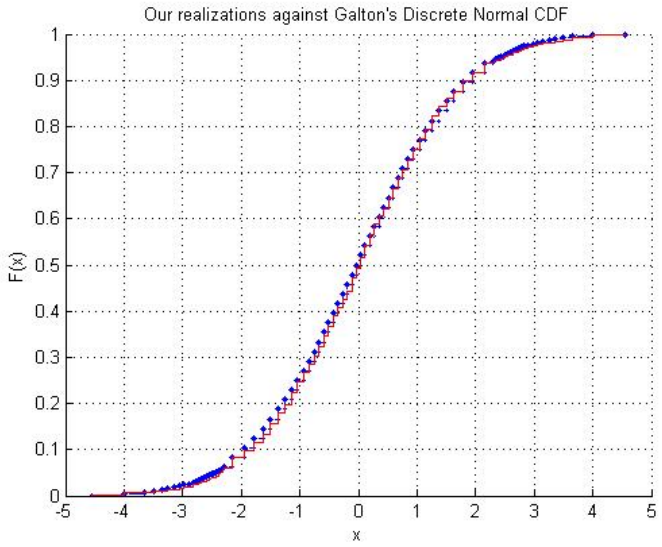


Figure 1.3: Plot showing the empirical DF of our results against GDN. Our values take on a stair case appearance and are very close to GDN. The main deviations occur mostly in the tails.

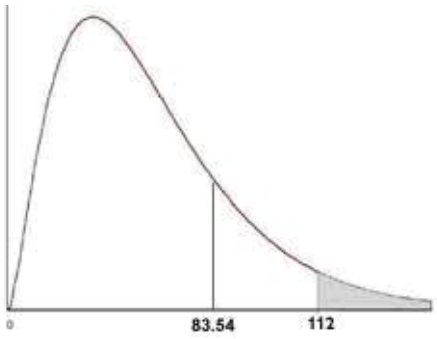
Chi Squared Test

A Chi Squared test was then performed on the data and the results[1] are summarised below.

[1]For the full table, please see Appendix A.

	Data Values	Observed Count	Expected Count	$(O-E)^2/E$
	-4.55	5	5.046875	0.000435372
	-4	7	5.046875	0.755853328
	-3.65	5	5.046875	0.000435372
	3.49	6	5.046875	0.180001935
	...	...	...	...
	3.65	8	5.046875	1.727989551
	4	11	5.046875	7.022107198
	4.55	5	5.046875	0.000435372
Total		1938	1938	
Chi Test Result				83.54798762

$$T = \Sigma_{i=1}^{90} \frac{(Obseverd - Expected)^2}{Expected} = 83.548$$



Conclusion

We cannot reject H0 at = 0.05 because the observed test statistic is outside the rejection region. In relation to our statistical question, this means that there is insufficient evidence to suggest that our sample is not from GDN.

Potential Modification

Since the standard normal distribution is more common in all areas, we wanted to convert Galton's Dice into a new set which can be used for simulating the standard normal distribution.

In his experiment, Galton took the mid-point of each probability interval, and then found the corresponding x- values. Instead of applying a tedious calculation to find the x-values, we took a z-value table, and found

the corresponding z-values to the upper bound of those intervals. This enables the creation of two new dice[2]:

Dice (1)	0.05	0.10	0.15	0.21	0.27	0.32
	0.37	0.43	0.49	0.55	0.61	0.67
	0.74	0.81	0.89	0.97	1.05	1.15
	1.26	1.38	1.53	*	*	*
Dice (2)	1.56	1.58	1.60	1.62	1.65	1.68
	1.70	1.73	1.76	1.79	1.83	1.86
	1.90	1.94	1.99	2.04	2.09	2.15
	2.23	2.31	2.42	2.56	2.80	4.00

Through `Matlab`, we were able to map the data gathered during our original experiment into the values shown in previous table, corresponding to the standard Normal, and develop the following plot:

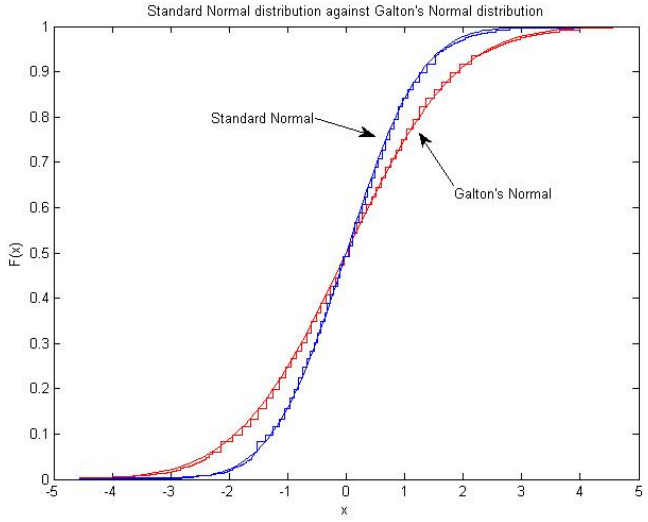


Figure 1.4: Plot showing the Standard Normal Distribution against Galton's Normal Distribution.

[2]Tables showing the new values for dice 1 & 2. The third dice can remain the same as Galton's.

Author Contributions

Brett - Constructed dice, gathered majority of the data results, constructed report, conducted spell/grammar check.

Joe - Wrote up `Matlab` code to analyse and plot data, entered in data results, constructed presentation and discovered a modification to Galton's experiment.

References

Dice for Statistical Experiments. *Nature* (1890) Vol 42, No 1070

Stochastic Simulation in the Nineteenth Century. *StatisticalScience* (1991) Vol 6, No 1

http://www.galton.org

http://www.anu.edu.au/nceph/surfstat/surfstat-home/tables/normal.php

Appendix A

	Data Values	Observed Count	Expected Count	$(O-E)^2/E$
	-4.55	5	5.046875	0.000435372
	-4	7	5.046875	0.755853328
	-3.65	5	5.046875	0.000435372
	-3.49	1	5.046875	3.245017415
	-3.36	3	5.046875	0.830156734
	-3.25	3	5.046875	0.830156734
	-3.15	3	5.046875	0.830156734
	-3.06	4	5.046875	0.217153638
	-2.98	4	5.046875	0.217153638
	-2.9	3	5.046875	0.830156734
	-2.83	8	5.046875	1.727989551
	-2.77	5	5.046875	0.000435372
	-2.72	3	5.046875	0.830156734
	-2.68	3	5.046875	0.830156734
	-2.64	3	5.046875	0.830156734
	-2.59	6	5.046875	0.180001935
	-2.55	4	5.046875	0.217153638
	-2.51	5	5.046875	0.000435372
	-2.47	4	5.046875	0.217153638
	-2.43	6	5.046875	0.180001935
	-2.39	10	5.046875	4.861116486
	-2.35	6	5.046875	0.180001935
	-2.32	8	5.046875	1.727989551
	-2.29	6	5.046875	0.180001935
	-2.15	47	40.375	1.087074303
	-1.95	28	40.375	3.792956656
	-1.78	34	40.375	1.006578947
	-1.63	33	40.375	1.347136223
	-1.5	45	40.375	0.529798762

	Data Values	Observed Count	Expected Count	$(O-E)^2/E$
	-1.37	46	40.375	0.783668731
	-1.25	37	40.375	0.282120743
	-1.14	48	40.375	1.44001548
	-1.04	48	40.375	1.44001548
	-0.94	35	40.375	0.715557276
	-0.85	34	40.375	1.006578947
	-0.76	34	40.375	1.006578947
	-0.67	41	40.375	0.009674923
	-0.59	49	40.375	1.84249226
	-0.51	37	40.375	0.282120743
	-0.43	44	40.375	0.325464396
	-0.35	33	40.375	1.347136223
	-0.27	36	40.375	0.474071207
	-0.19	36	40.375	0.474071207
	-0.11	55	40.375	5.297600619
	-0.03	38	40.375	0.139705882
	0.03	45	40.375	0.529798762
	0.11	53	40.375	3.947755418
	0.19	48	40.375	1.44001548
	0.27	40	40.375	0.003482972
	0.35	35	40.375	0.715557276
	0.43	32	40.375	1.737229102
	0.51	42	40.375	0.065402477
	0.59	41	40.375	0.009674923
	0.67	46	40.375	0.783668731
	0.76	35	40.375	0.715557276
	0.85	38	40.375	0.139705882
	0.94	45	40.375	0.529798762
	1.04	44	40.375	0.325464396
	1.14	43	40.375	0.170665635

	Data Values	Observed Count	Expected Count	$(O-E)^2/E$
	1.25	55	40.375	5.297600619
	1.37	38	40.375	0.139705882
	1.5	35	40.375	0.715557276
	1.63	32	40.375	1.737229102
	1.78	42	40.375	0.065402477
	1.95	33	40.375	1.347136223
	2.15	40	40.375	0.003482972
	2.29	3	5.046875	0.830156734
	2.32	4	5.046875	0.217153638
	2.35	3	5.046875	0.830156734
	2.39	4	5.046875	0.217153638
	2.43	6	5.046875	0.180001935
	2.47	5	5.046875	0.000435372
	2.51	3	5.046875	0.830156734
	2.55	8	5.046875	1.727989551
	2.59	1	5.046875	3.245017415
	2.64	7	5.046875	0.755853328
	2.68	5	5.046875	0.000435372
	2.72	4	5.046875	0.217153638
	2.77	4	5.046875	0.217153638
	2.83	6	5.046875	0.180001935
	2.9	5	5.046875	0.000435372
	2.98	5	5.046875	0.000435372
	3.06	6	5.046875	0.180001935
	3.15	5	5.046875	0.000435372
	3.25	4	5.046875	0.217153638
	3.36	5	5.046875	0.000435372
	3.49	6	5.046875	0.180001935
	3.65	8	5.046875	1.727989551
	4	11	5.046875	7.022107198

	Data Values	Observed Count	Expected Count	$(O-E)^2/E$
	4.55	5	5.046875	0.000435372
Total		1938	1938	
Chi Test Result				83.54798762

Appendix B

Table 1

0.03	0.51	1.04	1.78
0.11	0.59	1.14	1.95
0.19	0.67	1.25	2.15
0.27	0.76	1.37	*
0.35	0.85	1.50	*
0.43	0.94	1.63	*

Table 2

2.29	2.51	2.77	3.25
2.32	2.55	2.83	3.36
2.35	2.59	2.90	3.49
2.59	2.64	2.98	3.65
2.43	2.68	3.06	4.00
2.47	2.72	3.15	4.55

Table 3

++++	+–+	–++	+-+
+++-	+—	–+-	+–
++-+	-+++	—+	-++
++–	-++-	—-	-+-
+-++	-+-+	+++	–+
+-+-	-+–	++-	—

Chapter 2

A General Dynamic Model For the Rat Population in Haast

For use in mammalian pest control in New Zealand conservation lands

Howie FU

and Lin WANG

Abstract

Survival of some native bird species in New Zealand requires long-term control of pest populations (e.g. stoats, mice and rats). Before developing strategies to control pest populations, it would be useful to estimate the underlying pest population sizes. New Zealand's Department of Conservation is interested in ecological management. Our project focuses only on the rat population under treated conditions, namely that the stoats, which prey on rats, have been eradicated in Haast, an experiment site located in the Sounth Island of New Zealand.

Introduction

In New Zealand, rats are abundant and widespread, and they are a great threat to conservation. Rat predation is an important factor in the continued decline of several bird species, such as brown kiwi, black stilt, New Zealand dotterel, kaka, yellow-crowned kakariki and yellowhead[1]. The Department of Conservation (DOC) is interested in managing the rat population in New Zealand forests. Estimating the rat population size is the first step toward population management. DOC periodically sets up many tunnels inside the forest and counts the fraction of tunnels that were visited by at least one rat in a fixed duration of time. We can estimate the rat population sizes from this tunnel-track data. Our project involves finding the null

[1] 1. Progress in mammal pest control on New Zealand conservation lands $SCIENCEFORCONSERVATION$127 Published by Department of Conservation, P.O. Box 10-420 Wellington, New Zealand.

distribution for the proportion of the tunnels visited using parametric bootstraps of N-simple random walkers for a given duration over an appropriate 2D grid.

Materials & Methods

DOC set up 15 lines (150 tunnels) inside an area. Two areas were ivestigated, one of 10,000 ha and one of 14,000 ha. All tunnels are placed along a line 450m long, with 10 tracking tunnels in each line. Tunnels are 50m apart and lines always run North-South, and the minimum spacing between any two lines is 1km. Inside the tunnel, they slide a tray with two papers (one at each end) and an ink-pad at the centre. They place bait (peanut butter or rabbit meat) at each end and rats walk through leaving footprints on the papers. Then they score tunnels as either 'tracked"or "untracked"by rats. Overall, scores for each site are used as an index of mammal abundance. The index is calculated from the number of "tracked"tunnels divided by the total number of tracks. The index indicates rat densities during the observation period. Only two possible outcomes exist for the data - "tracked"and 'untracked"- no matter how many times a tunnel is visited. Therefore, to set our model, we can denote those two possible outcomes as 0 and 1, respectively.

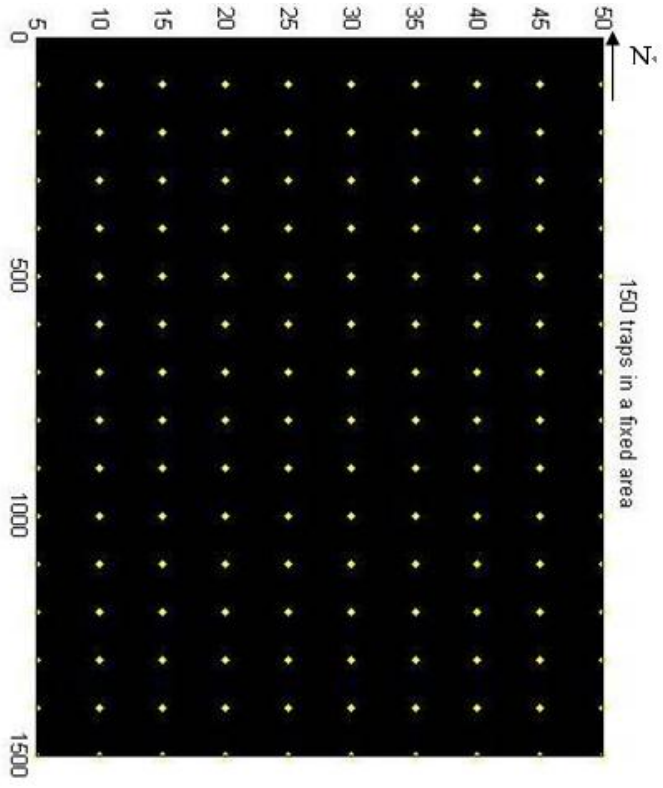


Figure 2.1: The initial model for how the traps were set.

Statistically, random walking uses a Markov chain model, where a rat can pick any one of four directions

at random. The rat keeps walking, having a chance of 1/4 for each direction at every step. Traps are set following a Bernoulli RV where the number of 0s and 1s accumulate. Therefore, we are able to estimate how many traps have been visited by counting the resulting 1s. We generate 100 data sets (tunnels visited) for a certain number of rats. By using the bootstrap method, we can get the mean value for each data set. The more data sets we generate, the more accurate our estimation will be. We observe a 95% confidence interval for our data sets. Once we have calculated a range of mean values, we can find out the relation between the index and the rat population.

Results

Firstly, we make a model for mammals which live in the same hole (e.g. bats). They all come along the tunnel at some time randomly during the night.

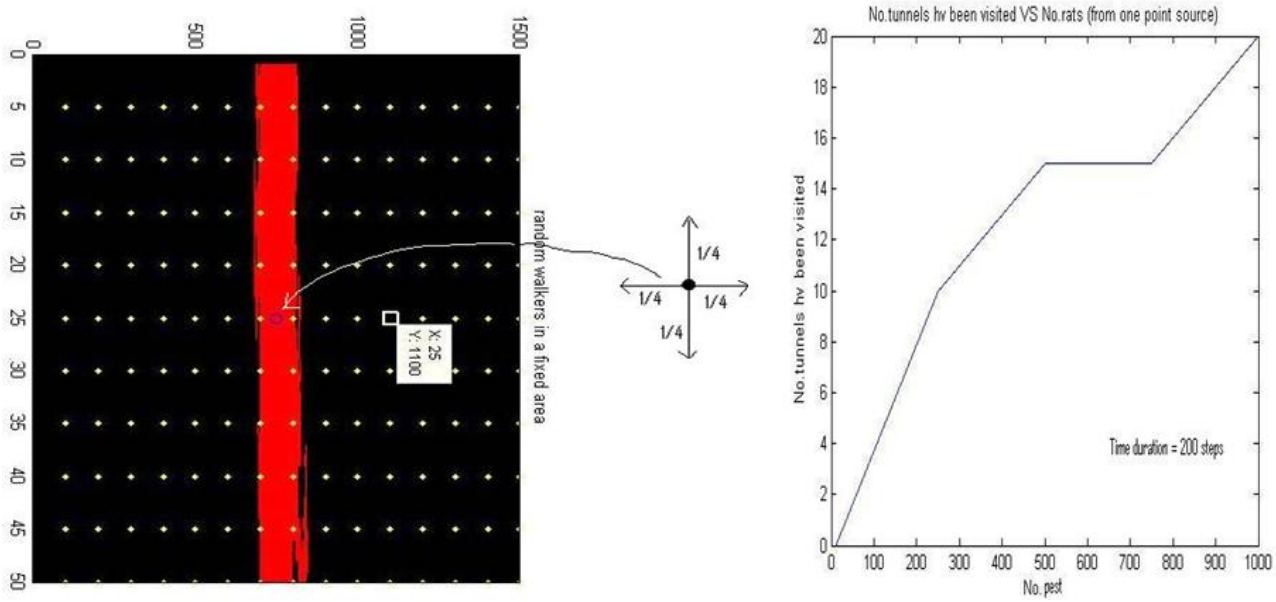


Figure 2.2:

The yellow dots represent one tunnel, the blue rings represent point sources and the red lines represent the rat tracks. Each tunnel is denoted by 1 until it is visited, when the designation changes to 0. After a certain time, we score the total number of tunnels visited then divide this by the total number of tunnels to get the index. The index indicates the percentage of the tunnel that have been visited.

We see that the animals will not cover many tunnels as they all come from the same point source.

But in reality, rats do not tend to live together. So we run our model again, with the rats coming from different point sources.

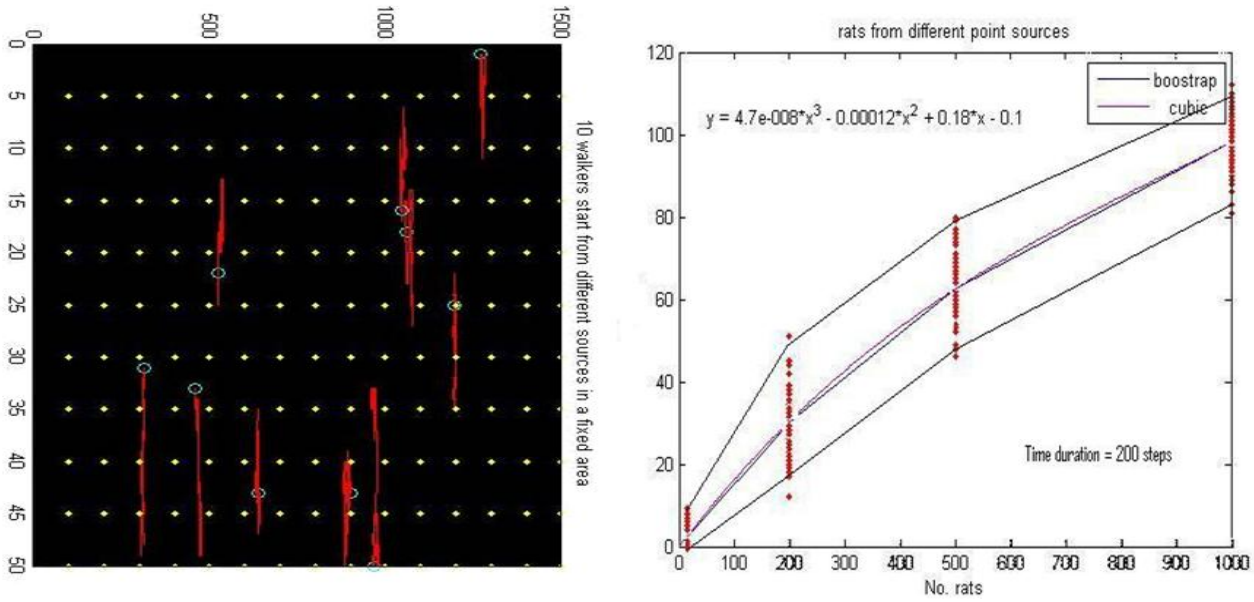


Figure 2.3:

Figure 2 shows the relationship between the number of tunnels and the number of rats. We used the bootstrap model to generate 100 data sets of Monte Carlo samples for 10 rats, 200 rats, 500 rats and 1000 rats. We observed the 95% confidence interval, which is the uncertainty of our model, and we connected the mean of each dataset. By adding a polynomial fit line, we have a cubic equation of the rat population index. Now we can estimate the population if we know the index value by using our equation.

We applied our equation to the data (see appendix) supplied by DOC.

Figure 2 shows that in the long-term, rat densities are increasing steadily. The population decreases every summer and increases every winter, especially in 2003. Weather forecasts from the winter of 2003[2] are given below:

• **5 August**. Frosty morning in many places, severe inland South Island, eg. -5°C at Tara Hills.

•**6 August.** Large temperature range at Dunedin Airport with a frosty start of -5°C but an afternoon high of 16°C.

•**7 August.** Heavy rain in Fiordland and Westland.

•**9-10 August.** Frosts and fogs in the south and east of the South Island, e.g. -5°C at Dunedin (9th). Fog and low cloud is persistent in Canterbury and on the Kaikoura Coast, where daytime highs only manage single digits, eg. 5°C at Timaru (9th) and Kaikoura (10th). However, Kapiti reaches 17°C (10th) because of a foehn northeasterly flow.

————MetService

We can see that in the winter of 2003, it was comparatively warm with lots of rain and frost, which we believe to be the most suitable weather for rats to live in.

[2]http://www.metservice.com/default/index.php?alias=2003winter0192994

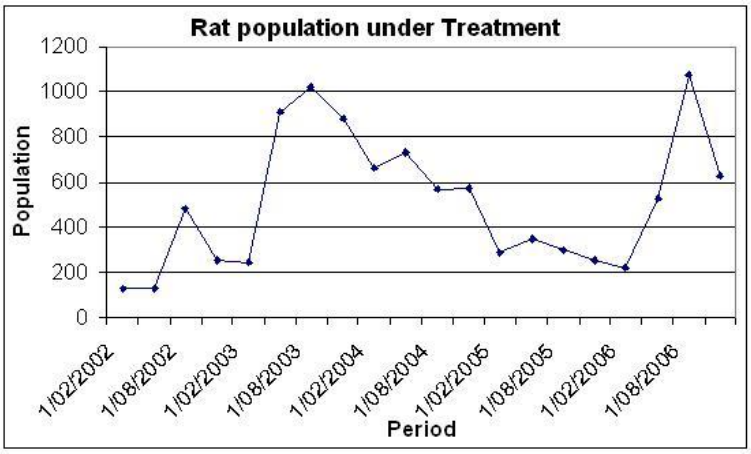


Figure 2.4:

Overall, the rat population seems to undergo a cycle. This may be a time series problem, but we need more data to verify if a clear pattern emerges.

Notice in our model, we assume the average distance for rats to walk in a night is $\sqrt{200} * 10\text{m} = 140\text{m/night}$.

Conclusion

According to our result, we cannot see any proof that the rat population increases when rats are not being preyed upon by stoats. Therefore, we believe that stoats are not the major factor that affects the rat population; something else must affect the rat density. However, we know when the peak density occurs in Haast so we can decide on some strategies to reduce the rat population. Some management recommendations can be found online. For example, it is possible to use poisoned dead mice for rat control, but this might kill other animals living in the forest or pollute the forest. Other strategies exist for reducing the rat population but we need to consider the environment as well.

Further Implementation

We could get data for the distance a rat walks in a night to make our model more accurate.

Establishing an interaction model would be useful for showing the effect of two or more dependency variables (e.g. stoats, mice and rats in one environment).

Author contributions

Many thanks to our lecturer Dr. Raazesh Sainudiin, Dr. Britta Basse and Dr. Ian Flux (NZ Department of Conservation). They spent their time to share and discuss the information with us and gave us a chance to analyse this interesting data collected by Dr. Flux.

Lin WANG did most of the `Matlab` coding part. Howie FU wrote this report and did the slides for our presentation.

Appendix

Haast Treatment	
Target Date T	Rats T%
1/11/2001	
1/02/2002	13.78
1/08/2002	40.48
1/11/2002	24.83
1/02/2003	24.22
1/05/2003	61.62
1/08/2003	68.56
1/11/2003	60.11
1/02/2004	50.26
1/05/2004	53.63
1/08/2004	45.38
1/11/2004	45.71
1/02/2005	27.81
1/05/2005	31.90
1/08/2005	28.44
1/11/2005	25.02
1/02/2006	22.00
1/05/2006	43.33
1/08/2006	75.56
1/11/2006	48.54

Chapter 3

Analysis of the distributions of Radiata pine circumferences from two different sites

Eli Thomas
and Jason Page

Abstract

In New Zealand, a major export is sawn Radiata pine. Milling usually takes place when the tree is 27yrs old and 35m in height. Because of the differing environments that these trees are planted in, it would be reasonable to assume that the age at which the trees reach this ideal height will vary, implying different rates of growth between environments. Therefore the null hypothesis ($\mathbf{H_o}$) for this experiment is that the distribution of tree circumferences (indicative of growth) from two supposedly different environments come from the same distribution.

Data were gathered from two Radiata pine plantations in Bottle Lake Forest. Statistical analysis of the collected data included non-parametric methods of bootstrapping and a permutation test. These tests concluded that at the 95% confidence level, the two sites came from the same distribution.

Introduction and Motivation

In New Zealand, a substantial industry is that of log exports. The main species of tree grown in this country and exported is Radiata pine, as it covers 89.2% of plantation forest area. Exports of Radiata pine, in sawn timber form, from this country last year (ending 30th of June 2007), accounted for $694,657,000 paid for 1,814,000 m3 of plantation forest according to the Ministry of Agriculture and Forestry (MAF) *Exports of Forestry Products* report. Logging of Radiata pine usually occurs when the tree is 27yrs old and at a height

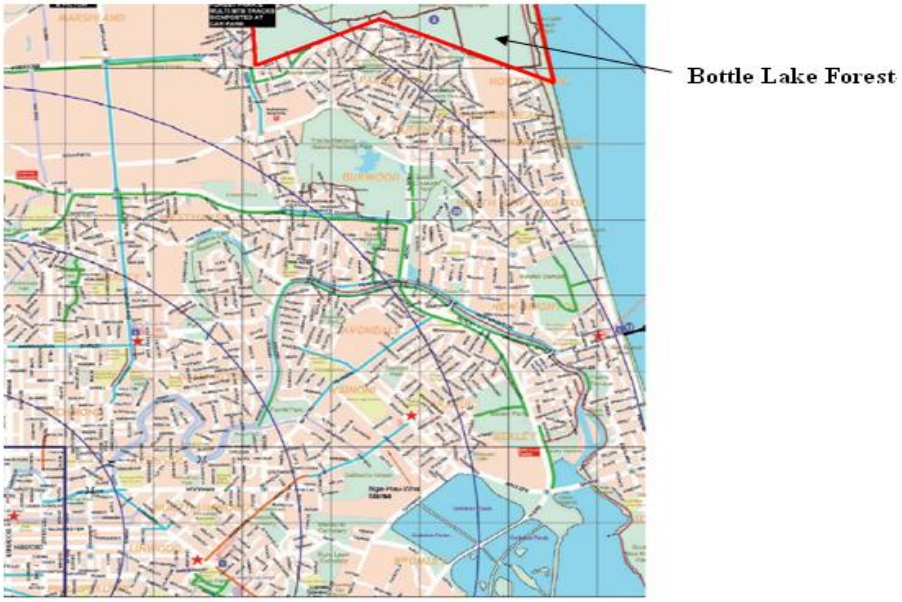


Figure 3.1:

of around 35m. One would assume that the rate at which the trees grow is dependent on the environment that they are in. Therefore, allowing for different growth rates and, in turn, different ages at which the tree will reach the preferred height for milling between these environments is more realistic. This assumption brings about the aim of this experiment, which is to determine if samples of Radiata pine from two different growing environments/sites do indeed come from the same distribution. To do this, we will be measuring the circumference of the tree, as trying to measure the height is impractical and tree circumference is also an indication of growth. The null hypothesis of this experiment is as follows:

$\mathbf{H_o}$: the distribution of circumferences at site A is equal to that at site B, (i.e. they come from the same distribution).

The alternate is:

$\mathbf{H_a}$: the circumferenices at the two sites do not have the same distribution.

To do this, we decided to take samples from two different sites of trees in the same plantation, namely Bottle Lake Forest.

Materials and methods

After choosing our two sample sites to be both within Bottle Lake Forest park in northern Christchurch, we then had to find two separate plantations that were planted at approximately the same time. Our two sites were chosen based on distance from the sea (as this was what we assumed would give a different growing

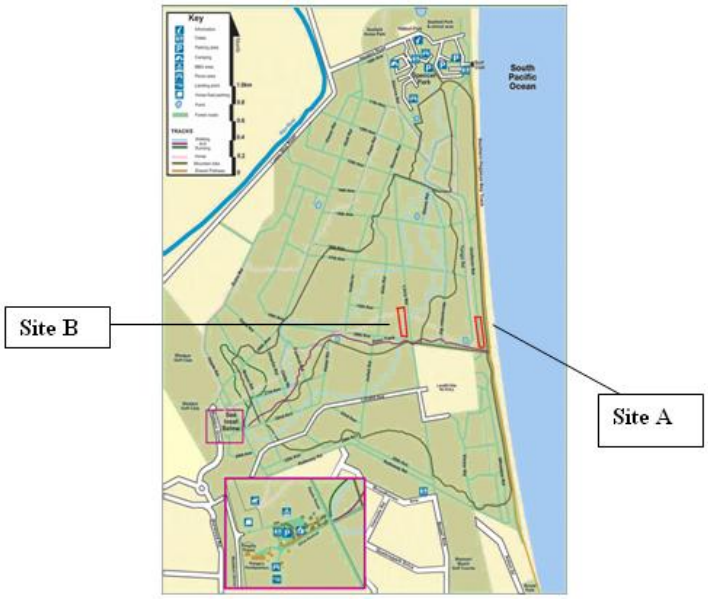


Figure 3.2: Bottle Lake Forest

environment). The sites are indicated on the map in Figure 3.2.

The reasoning behind these being two different environments is because we believed that the trees closer to the sea have a different soil type (more sand) and experience a different climate (because of shore breezes, and higher concentrations of salt in the soil and air from sea spray). To evaluate the distribution of circumferences at each site, we took 100 random circumference measurements. The experimental procedure that we undertook is detailed below:

•Starting at site A, which was 200m from the water line, with a 5m measuring tape capable of measuring in millimetres, we measured the girth widths of 100 randomly sampled distinct trees at breast height to the nearest mm. These case data were entered into a cell phone.

•Once 100 trees were sampled from site A, we repeated the measurements at site B. Site B was 1000m from the water line.

•We transferred the data to a personal computer and statistically analysed it using the `Matlab` program.

Statistical Methodology

Bootstrap

For the two samples, we computed the plug-in estimates of various summary statistics. These statistics include the mean, median and standard deviation. From this, we can gain a basic understanding of our samples. To gain a better understanding of our samples, we estimated the CDF with the ECDF algorithm

with a 95% confidence band. From the resulting graph, it should be relatively clear as to how the empirical distributions $\hat{F}^A$ and $\hat{F}^B$ are located relative to each other, and if they are both tending towards the same true DF F. This tendency will be indicated by significant overlap of confidence bands. This methodology relies on the Gilvenko-Cantelli theorem and the Dvoretzky-Kiefer-Wolfowitz inequality. We can get also get a $(1-\alpha)$ confidence interval for a point estimate, such as our plug-in estimate. (mean, median and standard deviation) via non-parametric bootstraps.

Permutation test

The permutation test is designed to determine whether the observed difference between the sample means is large enough to reject the null hypothesis Ho: that the two samples are drawn from the same distribution. First, the difference in means between the two collected samples is calculated (T(obs)). Then the observations of Sample1 (test site A) and Sample2 (test site B) are combined to form a new array. From this new array, 100 observations are sampled at random from it without replacement. The sample mean for these 100 observations is computed and the sample mean for the remaining 100 observations is also computed, and the difference between the resulting sample means is recorded. This process is repeated n times (e.g. 10,000) until a reliable estimation of the distribution is reached.

In this case, the purpose of the test is to try and reject the null hypothesis. The final p-value obtained allow us to interpret the strength of the evidence against the null hypothesis.

Results

Running the m-file ECDFplus.m computes the plug-in estimators (mean, median and standard deviation) for both sample sites.

	Site A (closest to beach)	Site B (furthest from beach)
Mean	1199.0 mm	1067.5 mm
Median	1232.5 mm	1067.5 mm
Std. Dev	522.6983 mm	318.0905 mm

From the plug-in estimates, site A contains more trees of a larger circumference, as its mean and median are both bigger than that obtained for site B, but it has a much larger standard deviation, indicating that the range of circumferences at this site is greater. Site B, on the other hand, has identical values for the median and mean, and a smaller standard deviation, showing that the trees there are more uniform in growth than those of site A.

`ECDFplus.m` produces a graph showing a plot of the ECDF of the sample values of both sites, with a $(1-\alpha)$ confidence band, in this case 95% for each value. The red line is the confidence band about sample A and the green line is the confidence band about sample B. It uses the Gilvenko-Cantelli theorem, which

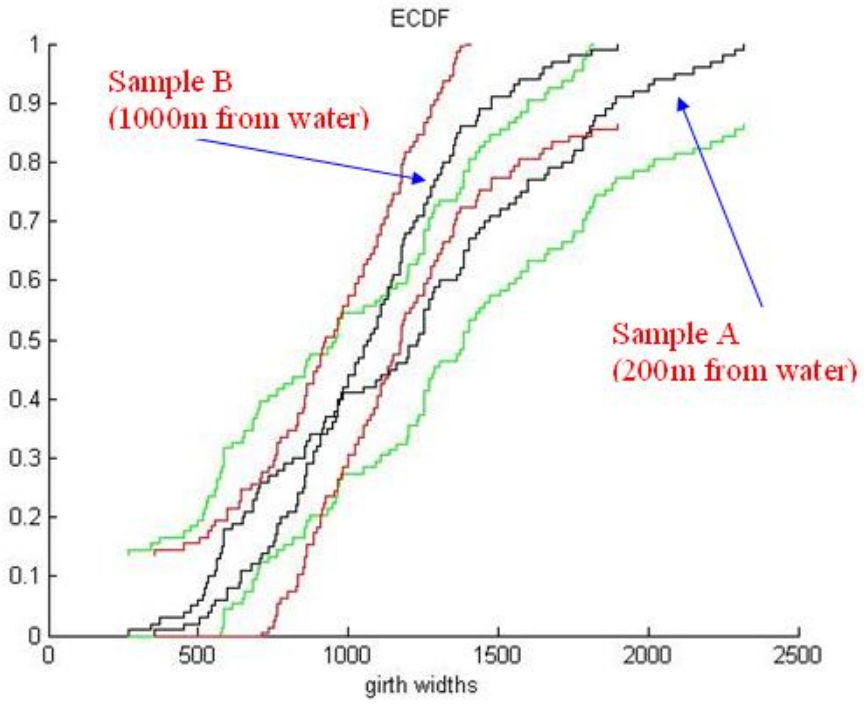


Figure 3.3:

states that as the sample size increases, $\hat{F}$ converges on the true DF F. The confidence bands of sites A and B show considerable overlapping, indicating that they are possibly tending towards the same distribution, at $= 0.05$.

Running the m-file bootstrap.m computes a confidence interval for the plug-in estimate of the mean and median for sample sites A and B at the 95% confidence level.

Confidence interval for site A mean = (985, 1325)

Confidence interval for site A median = (1097.4, 1302.2)

Confidence interval for site B mean = (975, 1147.5)

Confidence interval for site B median = (1006.9, 1130.3)

The confidence intervals for the mean and the median of both sites overlap, indicating that at the 95% confidence level, both samples are possibly from the same distribution, and that they both may have the same population mean and median. However, it pays to note that Site A has a much larger confidence interval than that of Site B because of the greater range of values obtained.

M-file `perm2.m` did a permutation test on the two samples to determine if $\hat{F}_A$ and $\hat{F}_B$ are equal. It did this by computing the absolute difference of the means on the initial sample arrays and on n generated ones. It then compares each difference generated with the n–array with the original difference to see if it is larger.

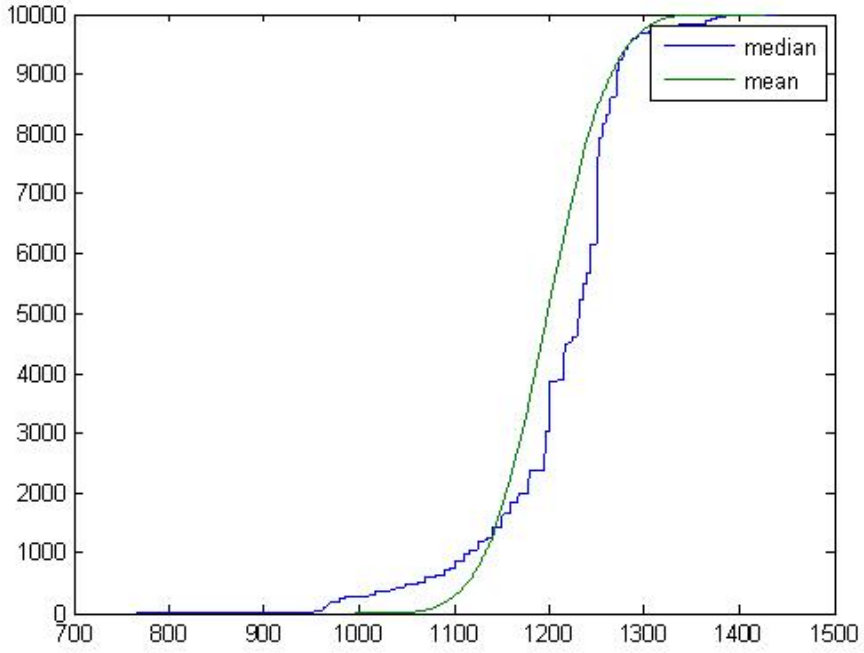


Figure 3.4: Bootstrap plot of mean/medians for Site A.

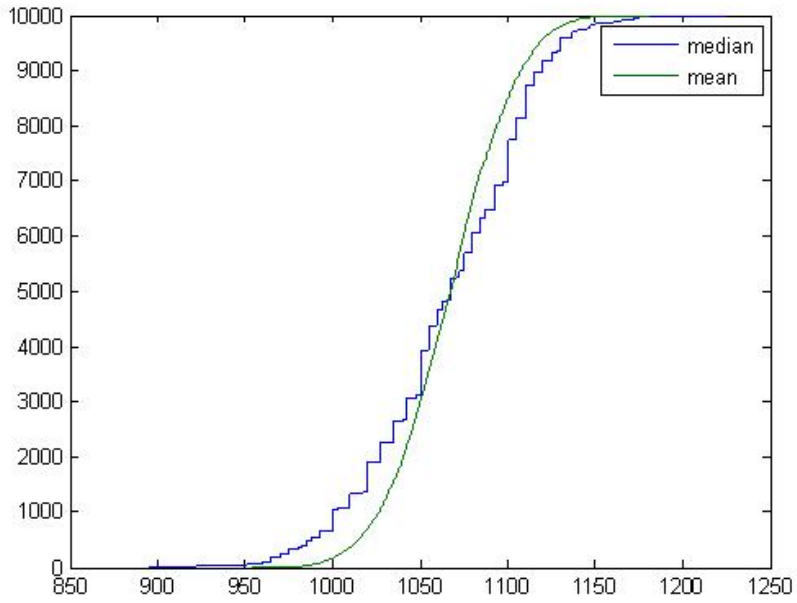


Figure 3.5: Bootstrap plot of mean/medians for Site B.

If this is so, it adds 1/n to the p-value. The p-value obtained for this data was 0.0361, which means that there is weak evidence to reject the null hypothesis. Even if the two population means are identical, we have a 3.61% chance of observing a difference as large as we did.

Conclusion

The null hypothesis Ho: that the distribution of site A's circumferences is equal to the distribution of site B, (i.e. they come from the same distribution) can not easily be rejected at the 95% confidence level. As evidence from the bootstrap method and the plot obtained from ECDF(2) both show, to varying degrees, the samples drawn from the two sites are indeed from the same distribution F. However, the permutation test result of 0.0361 shows that some evidence suggests that the $\mathbf{H_o}$ should be rejected, but as the evidence is weak and the other tests indicate that the true population means are the same, this leads us to not reject the $\mathbf{H_o}$. Increasing the sample sizes and number of samples taken is required to test the $\mathbf{H_o}$.

From the samples taken and the results obtained from them, we concluded that our chosen different environments had no effect or an insignificant effect on the growth of the trees, and this was reflected in the distribution of circumferences at the two sites. The other possible explanation is that our two environments were not different at all.

Author Contributions

Eli Thomas and Jason Page both have contributed equally during the process of this assignment. During the data collection stage, we both took and recorded the same amount of tree samples. Eli Thomas entered the data into `Matlab` to see at what the raw data looked like. Using a combination of functions given to us, those that were given and were later modified, and functions Jason Page and Eli Thomas wrote together, the results were obtained successfully. For the report, the Abstract and Conclusion were written by Jason Page and Eli Thomas together; Jason Page then wrote the Introduction/motivation and Results, with checking and editing from Eli Thomas. Eli Thomas then wrote Materials & methods and Statistical Methodology, with checking and editing by Jason Page. The slide show was made from equal parts by Jason Page, and Eli Thomas, and is a summary of our report.

Chapter 4

Diameter of *Dosinia* Shells

Guo Yaozong
and Shen Chun

Introduction

We collected some shells from New Brighton Pier that are commonly called *Dosinia anus* (Coarse Venus Shell). This species is a member of the class Bivalivia. Bivalivia lack a radula, and feed by filtering out fine particles of organic matter either from seawater (suspension feeders) or from surface mud (deposit feeders). In each case, food enters the mantle cavity in a current of water produced by cilia on the gills. Gills have a large surface area in relation to the size of the animal, and secrete copious amounts of slime-like mucus that not only traps the food particles but also acts as a lubricant for the passage of food to the mouth. In addition to having this feeding role, gills are the respiratory structures and are richly supplied with blood dorsally. Sexes are separate, although there is no external dimorphism. Gametes are shed into the seawater, where fertilisation occurs.

Unlike Venus shells from other parts of the world, this species has a flat disc-like shell. Found just below the low-tide mark along Brighton beach, it burrows just below the sand surface and feeds using two short, separate siphons. (*Life in The Estuary*, Malcolm B. Jones& Islay D. Marsden, Canterbury University Press, 2005).

Aim

To test whether the diameters of *Dosinia anus* shells on the north side of New Brighton Pier are identically distributed to those found on the south side of the pier.

Materials & Methods

We collected shells along the New Brighton beach to the left (north) and right (south) of the pier. We walked and picked up all the shells we could see, except broken ones. In about two and a half hours, we collected about two buckets of shells from each side of the pier (i.e. two from the left and two from the right).

After washing, drying and classifying the shells, we found that 254 of them were *Dosinia anus*, 115 collected from north of the pier and 139 from the southern. Then we used mechanical pencils to sketch the outline of each shell onto graph paper and measured the diameter of each in units of millimetres. The way we measured them was from top to bottom (as shown below). After that, we entered the data into a computer, and estimated the empirical CDF as well as confidence bands.

Methodology

In order to test the null hypothesis that the shell diameters of our species are identically distributed on both sides of the pier, we applied the non-parametric permutation test.

By using the permutation test, we tested whether the absolute difference between the two sample means were significantly different from each other.

Step 1: Observe value: T = X(left) - X(right)

Step 2: Combine [L1 L2 . . . . . L115 R1. . . . . R139] '254 Data'

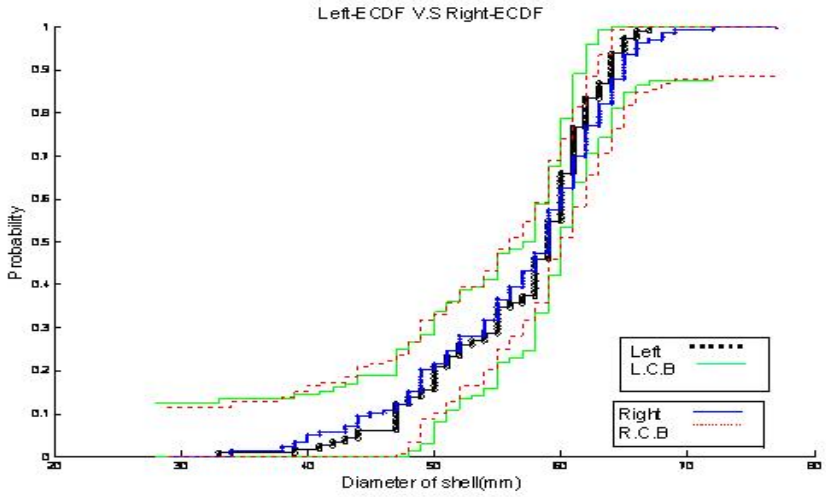
Step 3: Rearrange (`MATLAB` function: 'randperm'):

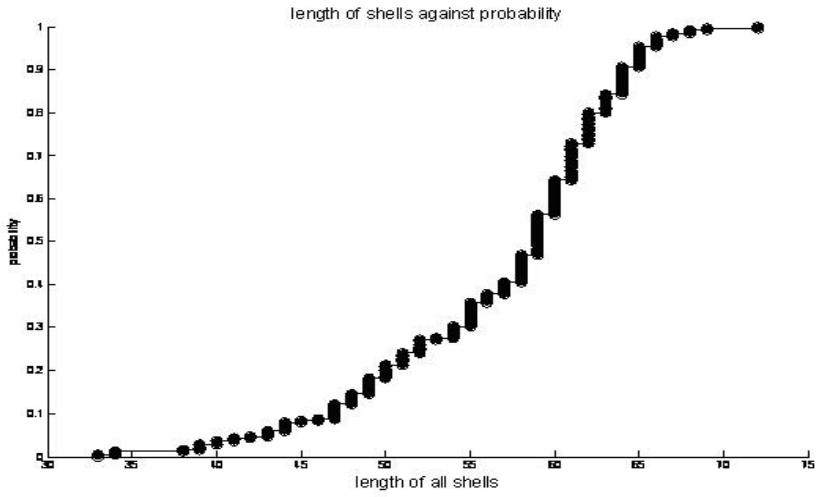
$$\begin{array}{ccccc} [\text{R110, L45, L78, R2}\ldots & | & \ldots\ldots\ldots\text{L20}] & & \text{'254 Data'} \\ \downarrow & & \downarrow & & \\ |\text{Mean(1-115)} & -- & \text{Mean(116-254)}| & = D_i(\text{recorded}) & \\ (i = 1, 2, 3 \ldots 10000) & & & & \end{array}$$

Step 4: Repeat Step 3 10000 times

Step 5: Find out how often 'Di' is greater than 'T', then divided this value by **10,000**. This is our **P-value**.

Result





Hypothesis testing

abs('mean for north'-'mean for south'): $|56.8173 - 56.6462| = 0.1711$(observed value) H_o: No difference can be observed between north and south H_a: A difference can be observed Alpha = 0.05

In the test, we found 8470 numbers were greater than 0.1711, so P-valve = 8470 / 10000 = 0.847

Conclusion

Since p-value is large, we do not reject the null hypothesis, as we do not have enough evidence to say there is a diameter difference in the distribution of *Dosinia anus* diameters between the north and south sides of the pier in New Brighton Pier.

Author contributions

Shen Chun and Yaozong Guo did all the work together.

Chapter 5

A Case Study of the Student Permit Car Park outside the Mathematics and Computer Science Building

ZHU Bo

and Xia Yinlong

Abstract

Parking at University of Canterbury is one of the challenges that a student has to face. A large number of students drive to the University but only a small proportion of parking spaces are available. Thus, our project aims to determine how hard it is to find a parking space at the student permit car park in front of the Mathematics and Computer Science building between 9 a.m. and 12 noon.

Introduction

Cars are the most popular mode of transportation around the world, and an increasing number of university students own cars. Student ownership is increasing at a faster rate than parking space on campus.

The University of Canterbury has 20,824 students enrolled in 2007 but only 1537 student car parking spaces are available on campus, which is 7.38 percent of the number of students. In this project, we were interested in the car park in front of the Mathematics and Computer Science building during the period of 9 a.m. to 12 noon on weekdays. There are 60 student permit parking spaces at the car park and everyone with a parking permit goes to the student permit parking area and searches for a parking space. If all 60 parking spaces are occupied then a driver has to wait for a car to leave, or leave the parking lot for another one on campus.

Moreover, the hypothesis of our project is that you can hardly find a car space at the car park in front

of the Mathematics and Computer Science building in the morning.

Materials and Methods

To test the hypothesis, we collected data to find the searching time of each car trying to find a parking space in the front of the building between 9 a.m. to 12 noon.

Data collection is an important part of the project and we tried to find a place from which we can overlook the car park in front of the Mathematics and Computer Science building. From the meeting room on the fourth floor of the Mathematics and Computer Science building, we could see all the entrances and exits to the car park, as well as all the parking space. From this meeting room, we recorded the time when a car entered the parking lot from any entrance in seconds. We assumed all the cars that entered the student permit parking area had a valid student permit and were trying to find a parking space there. We also recorded the time when each car left the car park. Thus, we can calculate the searching time by subtracting the entry time of a car from the leaving time of that car. The searching time is simply the duration a car spends in a filled parking lot while trying to find a parking space.

Statistical Methodology

Nonparametric estimation and the nonparametric bootstrap method were used in our case study to determine the 95% confidence interval of searching time of the student permit car park. In our experiment, we took 1000 bootstrap samples and calculated the mean of each sample, then calculated the 0.025 quantile and 0.975 quantile to yield the 95% confidence interval. The Maximum Likelihood Estimator was computed to fit the original data, where the MLE is$\hat{\lambda} = \frac{1}{sample\ mean}$.

Analysis of Results

Figures 1 shows the empirical CDF of the time of searching in the student permit car with the 95% confidence interval bound and true CDF plot. The true CDF plot falls in the 95% confidence range and fits the data reasonably well. Its long tailed shape clearly indicates that some unusual behaviours exist, which will be explained later.

Ninety-seven percent of people took between 0 seconds and 70 seconds to search for a space, whereas some people took more than 200 seconds. For example, there was one person took 700 seconds (i.e. approximately 12 minutes) to search for a space. This interesting fact also supports our hypothesis that it is extremely difficult to get a student permit car park space between 9:00 a.m. and 12:00 noon, and at least one student waited for over 12 minutes for a parking space.

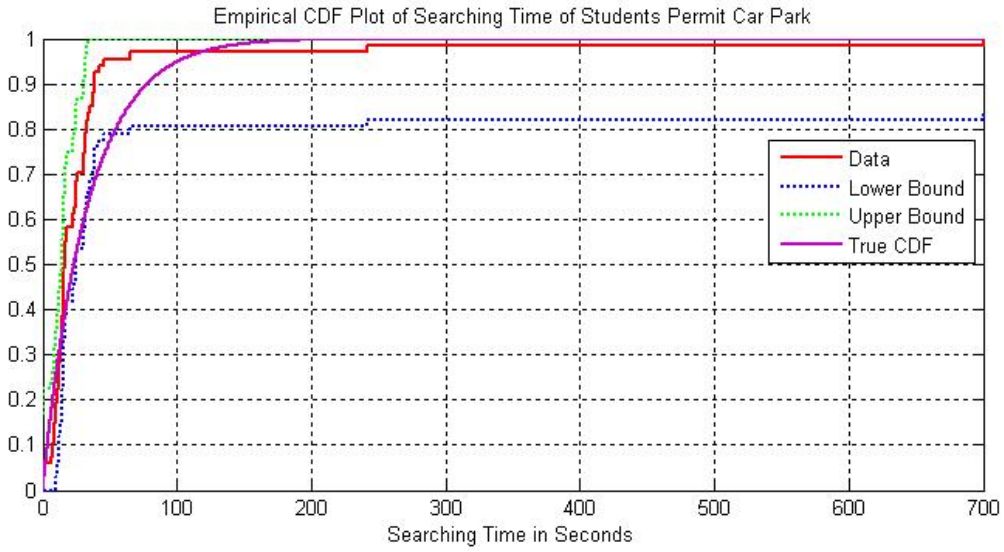


Figure 5.1: Empirical CDF plot of searching time of students permit car park.

Based on these data, we generated the 95% confidence interval by using the nonparametric bootstrap method of 1000 samples:

[18.7164, 57.5612]

This means that it took approximately between 19 seconds and 58 seconds to search for a parking space. But this result may not represent the true state of the searching time.

By removing unusually long searching times from the original data, our new empirical CDF plot is shown in Figure 5.2 below. The true CDF plot falls in the 95% confidence range and fits the data reasonably well. This graph represents the searching time distribution in a more accurate state. This may give a better indication of how long will it take to search for a space.

Based on the new data, the following 95% confidence interval is generated by using the nonparametric bootstrap method of 1000 samples:

[17.2932, 23.3404]

This means that it took approximately between 17 seconds and 23 seconds to search for a parking space. The searching time is very short. In fact, this is how long it takes for a car to enter the car park, pass all the spaces without stopping and leave the car park.

Furthermore, a total of 67 cars entered the student permit car park space during the three-hour observation period, but only 9 of them got parking spaces. This means approximately 87% of them did not get a parking space.

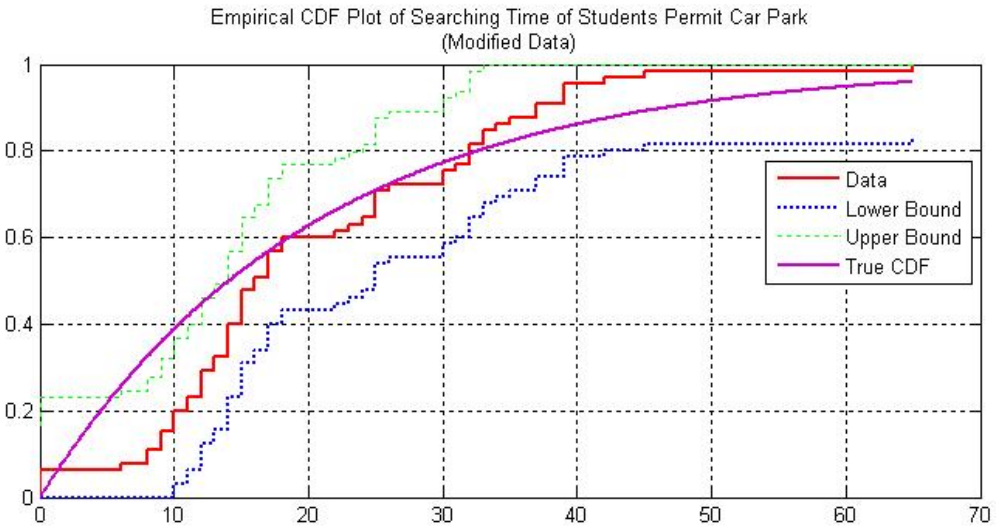


Figure 5.2: Empirical CDF plot of searching time of students permit car park (modified data).

Conclusion

Based on the nonparametric bootstrap experiment, results indicate that most people will not get student permit car park spaces during 9:00 a.m. and 12:00 noon outside the Mathematics and Computer Science building. Those who believe they could get a space if they work harder on searching for one will be disappointed.

Appendix

Matlab Codes

```
% This will draw the Time of Searching for a Student Permit Car Park
% Space with 95% Confidence Interval Bound

SampleSize = length(data);
% Get the x and y coordinates of SampleSize-based ECDF in x1 and y1 and
% plot the ECDF using the function ECDF2
[x1 y1] = ECDF2(data,0,0,0);
stairs(x1,y1,'r');

Alpha = 0.05; % set alpha to 5% for instance
Epsn = sqrt((1/(2*SampleSize))*log(2/Alpha)); % epsilon_n for the confidence band
hold on;
stairs(x1,max(y1-Epsn,zeros(1,length(y1))),'g'); % lower band plot
stairs(x1,min(y1+Epsn,ones(1,length(y1))),'g'); % upper band plot

%Plot the known true cdf
x = 0:0.1:max(data);
y = expcdf(x,mean(data));
```

```
plot(x,y)
hold off;

%Use the non-parametric bootstrap method to generate the 95% confidence
%interval by drawing 1000 bootstrap samples from original data.

ourSSample = sort(bootstrp(1000,@mean,data));
lowerq = qthSampleQuantile(0.025,ourSSample);
upperq = qthSampleQuantile(0.975,ourSSample);
ci = [lowerq,upperq]
```

Got no park	Got car park	Left
58	9	13
Total Arrival	Rate of Parked	
67	0.13	
	Rate of Not Parked	
	0.87	
	Searching Time	
Mean	0:00:34	

		Enter	Start Parking	Leave	Searching Time
Car	1	9:19:08	0:00:00	9:19:33	0:00:25
Car	2	9:19:30	0:00:00	9:19:55	0:00:25
Car	3	9:25:10	0:00:00	9:25:20	0:00:10
Car	4	9:30:08	0:00:00	9:30:23	0:00:15
Car	5	9:30:50	0:00:00	9:31:03	0:00:13
Car	6	9:30:03	0:00:00	9:30:12	0:00:09
Car	7	9:31:31	0:00:00	9:31:48	0:00:17
Car	8	9:33:18	0:00:00	9:33:27	0:00:09
Car	9	9:35:07	0:00:00	9:35:17	0:00:10
Car	10	9:38:28	0:00:00	9:38:40	0:00:12
Car	11	9:38:31	0:00:00	9:38:46	0:00:15
Car	12	9:40:55	0:00:00	9:41:10	0:00:15
Car	13	0:00:00	0:00:00	9:45:43	9:45:43
Car	14	9:46:06	0:00:00	9:46:41	0:00:35
Car	15	9:46:10	0:00:00	9:46:20	0:00:10
Car	16	9:47:07	9:47:07	0:00:00	0:00:00
Car	17	9:49:45	0:00:00	9:50:18	0:00:33
Car	18	9:51:38	9:51:50	0:00:00	0:00:12
Car	19	0:00:00	0:00:00	9:55:23	9:55:23
Car	20	9:55:52	0:00:00	9:56:10	0:00:18
Car	21	9:55:52	0:00:00	9:56:06	0:00:14
Car	22	9:57:16	0:00:00	9:57:33	0:00:17
Car	23	9:58:16	0:00:00	9:58:55	0:00:39
Car	24	0:00:00	0:00:00	9:58:38	9:58:38
Car	25	9:59:18	0:00:00	9:59:36	0:00:18
Car	26	9:59:32	0:00:00	9:59:38	0:00:06
Car	27	0:00:00	0:00:00	10:18:00	10:18:00
Car	28	10:18:00	10:18:30	0:00:00	0:00:30
Car	29	0:00:00	0:00:00	10:19:36	10:19:36
Car	30	10:29:10	10:29:10	0:00:00	0:00:00
Car	31	10:39:50	0:00:00	10:40:07	0:00:17
Car	32	10:45:45	0:00:00	10:46:02	0:00:17
Car	33	10:47:38	0:00:00	10:47:49	0:00:11
Car	34	0:00:00	0:00:00	10:48:29	10:48:29
Car	35	10:48:29	0:00:00	10:48:40	0:00:11
Car	36	10:49:01	0:00:00	10:49:17	0:00:16
Car	37	10:50:00	10:50:00	0:00:00	0:00:00
Car	38	10:50:19	0:00:00	10:50:33	0:00:14
Car	39	10:52:23	0:00:00	10:52:37	0:00:14
Car	40	10:56:00	0:00:00	10:56:12	0:00:12

		Enter	Start Parking	Leave	Searching Time
Car	41	10:56:57	0:00:00	10:57:09	0:00:12
Car	42	10:56:58	0:00:00	10:57:30	0:00:32
Car	43	10:57:09	0:00:00	10:57:31	0:00:22
Car	44	10:58:28	10:59:33	0:00:00	0:01:05
Car	45	0:00:00	0:00:00	10:59:23	10:59:23
Car	46	11:01:10	0:00:00	11:01:47	0:00:37
Car	47	11:01:37	0:00:00	11:02:09	0:00:32
Car	48	11:01:47	0:00:00	11:02:24	0:00:37
Car	49	11:02:03	0:00:00	11:02:18	0:00:15
Car	50	11:03:57	0:00:00	11:04:42	0:00:45
Car	51	11:05:22	11:06:04	0:00:00	0:00:42
Car	52	0:00:00	0:00:00	11:05:43	11:05:43
Car	53	11:05:48	0:00:00	11:06:21	0:00:33
Car	54	11:07:17	0:00:00	11:07:25	0:00:08
Car	55	0:00:00	0:00:00	11:07:26	11:07:26
Car	56	11:08:58	0:00:00	11:09:14	0:00:16
Car	57	0:00:00	0:00:00	11:14:41	11:14:41
Car	58	0:00:00	0:00:00	11:16:25	11:16:25
Car	59	11:16:35	0:00:00	11:16:35	0:00:00
Car	60	11:20:11	0:00:00	11:20:42	0:00:31
Car	61	11:24:23	11:24:53	0:00:00	0:00:30
Car	62	0:00:00	0:00:00	11:24:43	11:24:43
Car	63	11:26:13	11:26:45	0:00:00	0:00:32
Car	64	11:28:59	0:00:00	11:29:07	0:00:08
Car	65	11:31:20	0:00:00	11:31:45	0:00:25
Car	66	11:38:10	0:00:00	11:38:23	0:00:13
Car	67	11:39:47	0:00:00	11:40:11	0:00:24
Car	68	11:44:12	0:00:00	11:44:35	0:00:23
Car	69	11:47:01	0:00:00	11:58:41	0:11:40
Car	70	11:50:22	0:00:00	11:50:56	0:00:34
Car	71	11:52:34	0:00:00	11:53:13	0:00:39
Car	72	11:52:50	0:00:00	11:52:59	0:00:09
Car	73	11:55:36	0:00:00	11:55:50	0:00:14
Car	74	11:56:04	0:00:00	11:56:29	0:00:25
Car	75	11:56:37	0:00:00	12:00:39	0:04:02
Car	76	11:56:58	0:00:00	11:57:24	0:00:26
Car	77	11:59:43	0:00:00	11:59:57	0:00:14
Car	78	11:59:49	0:00:00	12:00:04	0:00:15
Car	79	0:00:00	0:00:00	12:02:15	12:02:15
Car	80	12:02:24	0:00:00	12:03:03	0:00:39

Table 5.1: Data

Chapter 6

Species counts of Bivalve shells in New Brighton Beach

WANG YuanCheng (James)
and HAN Dong (Winter)

Abstract

The probability of finding a particular species of Bivalve shell may vary. Ten species of the class Bivalvia are known to occur along the shores of Christchurch. They are Greenshell mussel, Ribbed mussel, Nesting mussel, Large trough shell, Triangle shell, Pipi, Tuatua, Cockle, Coarse Venus shell and Piddock. Our hypothesis is that all ten species are equally likely to be found on the northern (left) and southern (right) sides of New Brighton Pier.

Introduction

We used the collected count data and found the Bayes estimator $\hat{\theta_j} = E(\theta_j|y) = (a_j + n_j)/(a + n)$, where $\hat{\theta_j}$ is the posterior mean for a given species j, $a_j = 1$ as we assume all species are equally likely to be found (uniform Dirichlet prior); a is the sum of a_j and n is the total count of all the species found; and n_j is the number found for a given species j, where j takes a number from 1 to k, and $\Sigma_{j=1}^{k}\theta_j = 1$. In our case, k is 10 as we expected to find 10 species. By doing so, we wanted to answer the statistical question whether we are equally likely to find all species of shell on either side of New Brighton Pier.

Materials and Methods

Firstly, we selected one of the beaches in Christchurch. Furthermore, as New Brighton Pier can be treated as a landmark, we decided to collect shells from both sides of the pier and record the number found of each species.

Secondly, according to the reference book: "*Life in the Estuary: Illustrated Guide and Ecology*"(Malcolm B. Jones and Islay D. Marsden, published by Canterbury University Press 2005), we expected to find the following 10 species:

(1) Green mussel, common in many parts of New Zealand; (2) Ribbed mussel, often found in beds dominated by the green mussel, but never present in high numbers in the estuary; (3) Nesting mussel, quite rare in the estuary but can be easily recognised by its habitat; They like hiding in 'nests' of loosely woven strands of byssal material. (4) Large Trough shell, found at very low tide in the sand offshore of New Brighton Spit; (5) Triangle shells, found at the low-water mark on the surf beach on the New Brighton Spit, normally down to 6 m underwater; (6) Pipi, found at the end of Brighton Spit; (7) Tuatua, the dominant bivalve on the low tide on the exposed beach at New Brighton; (8) Cockle and (9) Coarse Venus shell, both of which can be found below the low-tide mark along Brighton beach. (10) Empty Piddock shells can be seen lying on the shore.

We tabulated the raw data and used the parametric bootstraps and re-sampling for multinomials 10,000 times by the de Moivre method with the expected probability being the same for both sides of the pier. From this, we decided whether to reject or accept the null hypothesis, based on the test statistic we used.

Statistical Methodology

	species	NL	NR	Total
1.	Greenshell mussel	3	4	7
2.	Ribbed mussel	0	0	0
3.	Nesting mussel	0	0	0
4.	Large Trough shell	4	6	10
5.	Triangle shell	94	152	243
6.	Pipi	14	16	30
7.	Tuatua	177	141	318
8.	Cockle	9	13	22
9.	Coarse Venus shell	115	139	254
10.	Piddock	0	0	0
	total count	413	471	884

Table 6.1: The raw data. NL refers to the number of shells found on the left side New Brighton Pier; NR refers to the number of shells found on the right side New Brighton Pier; Total refers to the total number of shells found on both side New Brighton Pier.

We firstly used the collected count data to calculate a Bayesian estimate $\hat{\theta_j}$ forθ_j, $\hat{\theta_j} = E(\theta_j|y) = (a_j + n_j)/(a + n)$. Recall that θ_j is the probability of finding a given species j. We set a_j=1 as we make the prior assumption that each species is equally likely to be found. Let $a = \Sigma_{j=1}^{10} a_j = 10$, and $n = \Sigma_{j=1}^{10} n_j$ is the total count of all the species found with n_j equalling the number found of a given species.

Next, we need to test the null hypothesis. ($\mathbf{H_o}$): the probability the number of shells found on the left

will equal those found on the right, for any given species, i.e. $F^L = F^R$ for the multinomial distribution with the estimated probability $\hat{\theta_1}$ up to $\hat{\theta_{10}}$.

To test this hypothesis, we used parametric bootstraps with multinomial re-sampling from an estimated $\hat{\theta}$ 10,000 times. Each time, we simulate 884, samples: 413 times from the left of the pier and 417 times from the right of the pier.

We use the simulated data to compute the test statistic:

$$t^* = \sqrt{\Sigma(E(\theta_j|y^{*\hat{(L)}}) - E(\theta_j|y^{*\hat{(R)}}))^2} = \sqrt{\Sigma_{j=1}^{10}(\frac{1+n_j^{*(L)}}{10+413} - \frac{1+n_j^{*(R)}}{10+417})^2}$$

where $n^{*(L)}$ is the species count for the bootstrapped data from the left and $n^{*(R)}$ is that for the right. Notice that both $n^(L)$ and $n^(R)$ are integer-valued vectors of length of 10.

Then, we plotted the Empirical Cumulative Distribution Function (ECDF) for the 10,000 points and found the critical value for a one-tailed test with $\alpha = 0.05$ for our null hypothesis. If the observed test statistic $t_{obs} = \sqrt{\Sigma(\hat{\theta_{obs}^{(L)}} - \hat{\theta_{obs}^{(R)}})^2}$ was to the right of the critical value then we would reject the null hypothesis.

Results

See in Table 6.2 and Figure 6.1.

	species	$\hat{\theta_{obs}^{(L)}}$	$\hat{\theta_{obs}^{(R)}}$	$\hat{\theta_{obs}^{(T)}}$
1.	Greenshell mussel	0.0095	0.0104	0.0089
2.	Ribbed mussel	0.0024	0.0021	0.0011
3.	Nesting mussel	0.0024	0.0021	0.0011
4.	Large Trough shell	0.0118	0.0146	0.0123
5.	Triangle shell	0.2175	0.3181	0.2729
6.	Pipi	0.0355	0.0353	0.0347
7.	Tuatua	0.4208	0.2952	0.3568
8.	Cockle	0.0236	0.0291	0.0257
9.	Coarse Venus shell	0.2742	0.2911	0.2852
10.	Piddock	0.0024	0.0021	0.0011
	total probability	1	1	1

Table 6.2: The posterior mean for 10 species found. $\hat{\theta_{obs}^{(L)}}$ refers the posterior mean on the left side of the pier; $\hat{\theta_{obs}^{(R)}}$ refers to the posterior mean on the right side of the pier; $\hat{\theta_{obs}^{(T)}}$ refers the posterior mean on the both sides of the pier.

Conclusion

From the figure above, we can tell that the observed value is in the "reject" region. We rejected the null hypothesis $F^{(L)} = F^{(R)}$.

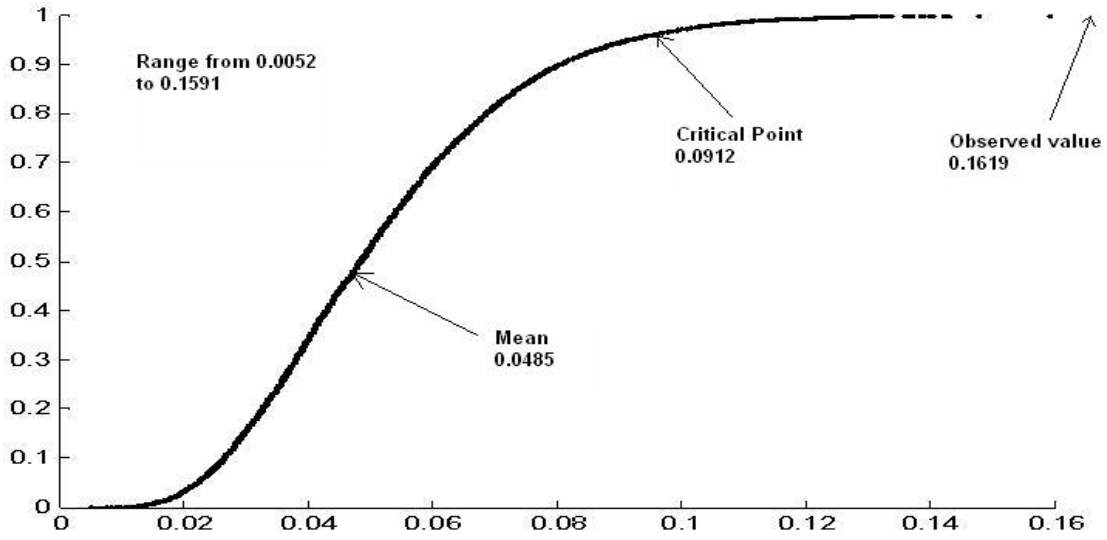


Figure 6.1: The ECDF of the Euclidean norm between the estimate θ for the left and the right sides of the pier.

We do not know why the probability of finding examples of these 10 species differs between the left and right sides of New Brighton Pier. It is possibly correlated with the local population densities of the ten Bivalve species. Neverless, the evidence shows that the probability of finding a given species differs, depending on which side of New Brighton Pier a search for shells is made.

Author contribution

Two people were involved in this project: YuanCheng, WANG and Dong, HAN. YuanCheng, WANG and Dong, HAN gathered and organised the data. The methodology was implemented by YuanCheng, WANG with help from Dong, HAN. YuanCheng, WANG and Dong, HAN wrote and edited the report together.

Chapter 7

Regressions on outcomes of progressively shaved dice

Russell Gribble

and Yuanqi Ye

Abstract

We wanted to find out how the probability of different faces of a dice varied as we progressively shaved off one side of the dice. We did this by tossing each dice 200 times and analysing the results.

Introduction

Intuitively, you would expect that a perfect cube would, when rolled, have each of its faces come up with equal probability, while a flat square would almost always land on one of its flat sides. Thus as fractions are shaved off a dice, we expect the flatter sides to land upwards more often when rolled. Our goal in this project was to determine how the probability changed as the dice were shaved off progressively.

Materials and Methods

Acquiring four dice with uniform density proved to be a challenge. Originally, we had considered using wooden blocks, but upon thinking about it, we realised that these would quite probably have uneven density, introducing an unwanted bias. Our next attempt was to use eraser rubber to produce four progressively shaved dice. However, we also had concerns about these as they were quite small. Lastly, we managed to have some hard foam cut. Originally, we attempted to have them cut by robot, but unfortunately this resulted in uneven sides, as the heated wire the robot used vaporised an unpredictable amount of foam. Eventually, we used the same foam cut by a knife. Even though it had some surface irregularities, we reasoned that the same amount of mass should be present, and it should not introduce a bias. The dice were cut with one dimension being 6/6, 5/6, 4/6 and 3/6 of the other dimensions for each cube sequentially. The size of the perfect cube dice was 40mm in all dimensions.

We gathered data by throwing each dice 200 times. We threw the dice upwards about 20cm, so as to attempt to ensure that all rolls were independent.

Statistical Methodology

We put the raw data into Excel for convenience. We then put it into `Matlab` for further analysis, and summed up the totals for each dice.

Performing a chi squared test on the regular cube against what we expected (i.e. a probability of 1/6 for each face), we failed to reject the null hypothesis that the data for the regular cube was unbiased at a 95% confidence level, and continued with the analysis.

From this point, we combined the results of numbers one and six coming up, as these were our 'flat' sides, and combined two, three, four, and five, as these were the other sides. We then obtained MLE estimates for the likelihood of the flat sides coming up for each dice.

Following this, we used the MLE estimates to form parametric bootstrap estimates for each dice so as to get a feel for the spread of our data. We also plotted 95% confidence intervals from these bootstraps.

Having acquired a feel for the likely spread of our data, we proceeded to plot scatter graphs of the MLE estimates with their 95% confidence intervals vs. the ratio of the area of the sides, to see if we could acquire a fairly close fitting regression line within them. We used SAS to do the regression as well as Excel, and inputted the extra data from SAS onto the plot done in Excel.

Results

Discussion

We used a chi squared test as below:

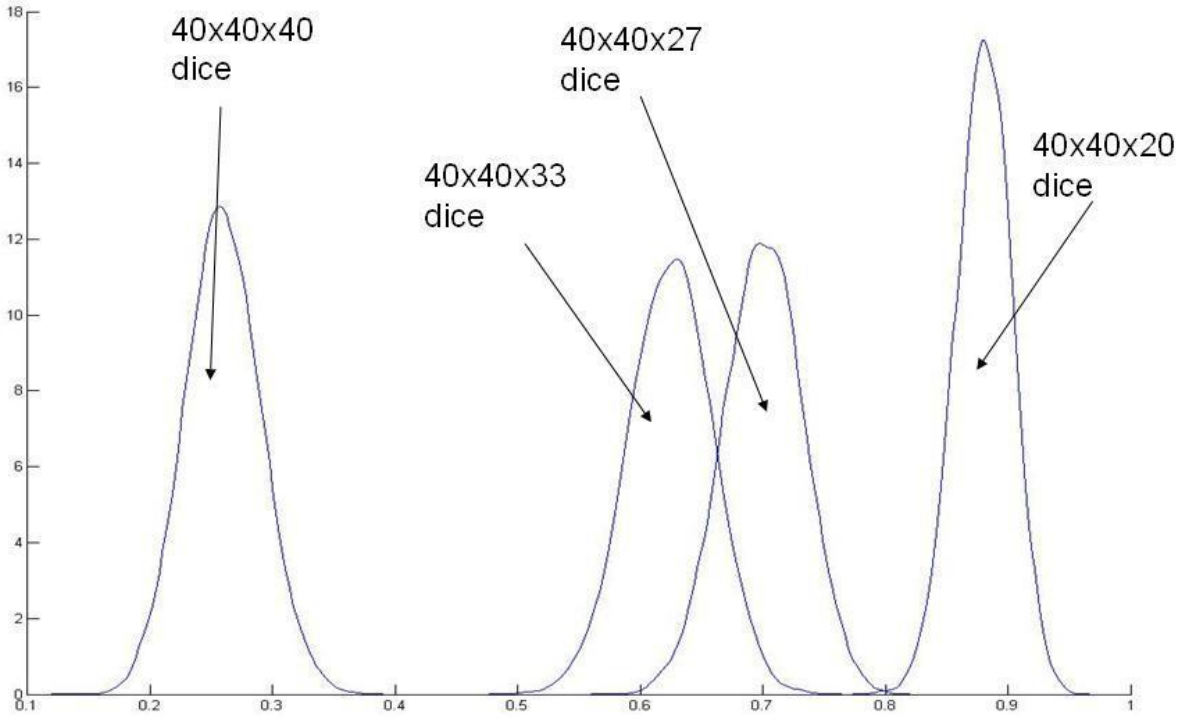


Figure 7.1: PDF plot of parametric bootstrap estimates (10,000 samples).

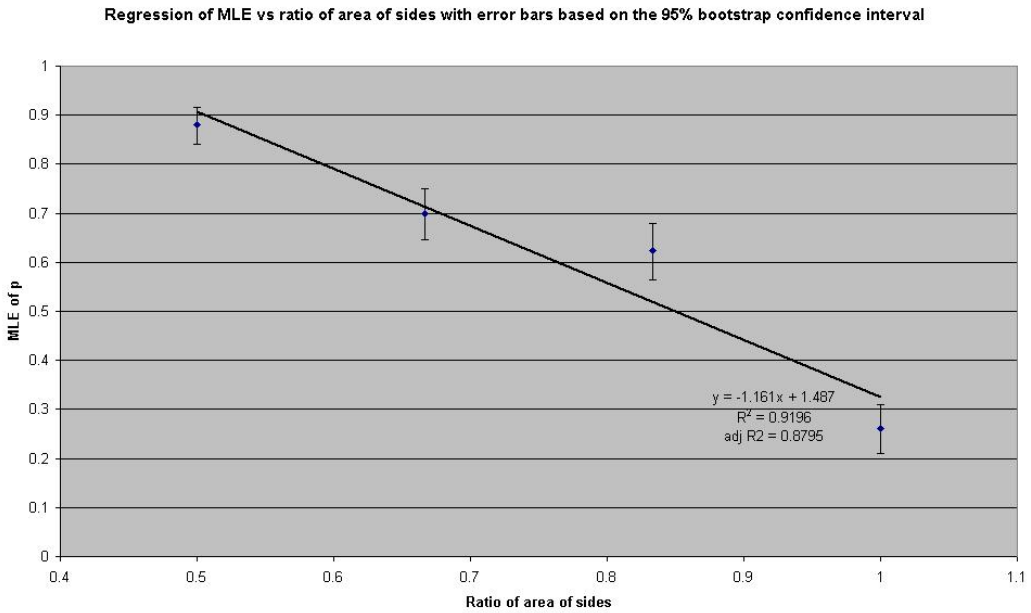


Figure 7.2: Regression of MLE vs ratio of area of sides with error bars based on the 95% bootstrap confidence interval.

Chi squared test:

Outcome	1	2	3	4	5	6
Data	26	31	37	44	36	26
Theoretical	33.3333	33.3333	33.3333	33.3333	33.3333	33.3333

We got the probability of the observed value as 0.1912 - thus failing to reject it at the 5% significance level.

We have significant concerns about the dice being biased, as the 95% bootstrap interval (0.21,0.31) for the dice we expected to be fair does not include 1/3, which theoretically should be the MLE. This makes us suspect that the dice we used were biased in some way, despite failing to reject the null hypothesis that the dice was fair in the chi squared test.

Conclusion

The best model that we were able to come up with is that the ratio of the flat side coming up is modelled by $1.487 — 1.161r$, where r is the ratio of the area of the sides.

However, as this model allows for probabilities of over 1, this is obviously not applicable to values outside the range of our experiment.

Author contributions

Dice throwing - Yuanqi Ye

Data entry - Russell Gribble

Analysis - Combined

Write up - Combined

Chapter 8

Estimating the Binomial probability p for a Galton's Quincunx

Bry Ashman

and Ryan Lawrence

Abstract

Galton's Quincunx is a physical device designed to simulate the discrete binomial distribution. we aim to create a physical model of the quincunx that is characterised by the probability of a ball going left is equal to the probability of it going right. From the conceptual model of the quincunx, we derive the binomial probability mass function. In order to evaluate the parameter of interest p, we will derive the maximum likelihood estimator and use this to estimate the actual parameter p of our physical model using 100 samples that are assumed to be independent and identically distributed.

Motivation

The binomial distribution is a fundamental discrete probability distribution, being the natural extension of the Bernoulli trial to the sum of Bernoulli trials. The distribution describes the number of successes in a sequence of n binary trials, with a probability p. Each of these trials is a Bernoulli trial parameterised by p. A binomial distribution parameterised by $n = 1$ and p is simply a *Bernoulli(p)* trial.

The quincunx was invented by Sir Francis Galton originally to demonstrate the normal distribution. The quincunx is simply an array of pegs spaced so that when a ball is dropped into a device, it bounces off the pegs with a prabability p of going right and a probability of *1-p* of going left. It bounces off n pegs before being collected in a bin at the bottom of the device.

To this end, we aim to create a quincunx ideally parameterised by $p = 0.5$ with $n = 20$. To verify this, we will use maximum likelihood estimation to test the null hypothesis that $p = 0.5$.

Materials and Methods

Construction of physical model

The physical model that we created consisted of nails arranged in a pattern on a sheet of wood that we hoped would achieve as close to the ideal probability of p=0.5.

Materials

- Plywood Sheet (1200 x 600 x 20mm)
- Perspex Sheets (1200 x 600 x 2mm and 550 x 800 x 2mm)
- Timber Strips (1200 x 25mm and 600 x 25mm)
- Nails (30 x 2mm)
- Chrome Balls (20mm)

Construction Details

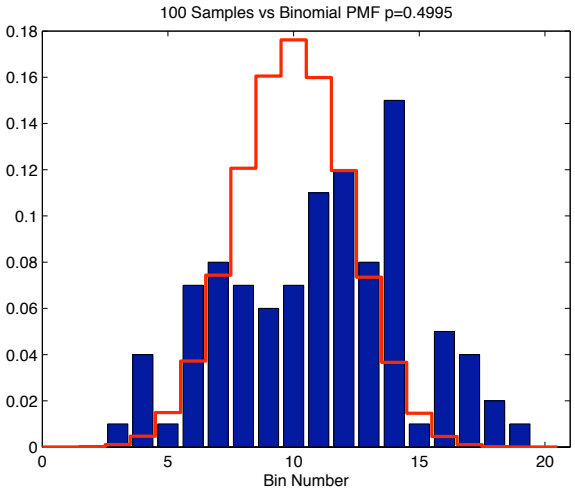
1. Mark 20 horizontal lines with 25mm spacings with the board in a portrait orientation.
2. Mark vertical lines at 25mm spacings from the centre of the board.
3. Place a nail at the top centre marking
4. Continue to place nails on the marked grid such that one marked grid point always separates the nails both vertically and horizontally.
5. Create the bins by attaching perspex strips directly below the nails of the last row.
6. Fit the edges to the main sheet.
7. The perspex sheet can now be attached to the edges of the quincunx.

A desirable feature of the quincunx is a release mechanism at the top to release the balls used to simulate a random variable and a release at the bottom to retrieve the balls after the experiment.

Sample Collection

To collect samples from the quincunx the balls are dropped into the device as identically as possible with sufficient time between each drop to ensure that the balls do not interfere with each other so as to keep the samples as identical as possible. The balls are collected in a series of bins numbered from 0 to 21, 0 representing the leftmost bin that the sample can be in and 21 being the rightmost bin. Since we assume

that each sample is identical and independent, we record the cumulative number of balls in each bin after dropping 100 balls. The data is shown in the blue bars in the next figure.



Statistical Methodology

Deriving the binomial distribution

The binomial distribution can be thought of as a random walk in one dimension. The parameters map to this model as p being the probability of taking a step right and $(1-p)$ the probability of taking a step left, and n being the total number of steps taken. From this, it follows that, for a given number of n steps, x of which are to the right and $n-x$ to the left, to find the probability that a combination of those n steps that will get you to the same point, you have to multiply the probability of the path by how many unique ways you can combine those steps. The number of ways of ordering the x right steps in a set of n steps is given by $\binom{n}{x}$. Therefore, the probability of ending up at a particular endpoint is as follows:

$$P(X = x) = \begin{pmatrix} n \\ x \end{pmatrix} p^x (1-p)^{n-x}$$

A note about the endpoint: I have used the convention that the leftmost bucket is 0. The end point numbers also tell you how many right steps you have in the quincunx.

Parametric Estimation

In order to estimate the parameter p for our physical model, we will use a maximum likelihood estimator (MLE) since it is often regarded as asymptotically optimal. However, for the binomial distribution, the MLE is equivalent to the Method of Moments.

Deriving the maximum likelihood estimator

$$L(p) = \Pi_{i=1}^{n} \begin{pmatrix} N \\ x_i \end{pmatrix} p^{x_i}(1-p)^{N-x_i}$$

$$L(p) = \Pi_{i=1}^{n} \begin{pmatrix} N \\ x_i \end{pmatrix} p^{\Sigma x_i}(1-p)^{nN-\Sigma x_i}$$

$$lnL(p) = \Sigma_{i=1}^{n} ln \begin{pmatrix} N \\ x_i \end{pmatrix} \Sigma x_i ln(p)(nN - \Sigma x_i) ln(1-p)$$

$$\frac{d}{dp} lnL(p) = \frac{\Sigma x_i}{p} - \frac{nN - \Sigma x_i}{1-p}$$

We can now set $\frac{d}{dp} lnL(p) = 0$ to find the maximum:

$$0 = \frac{\Sigma x_i}{p} - \frac{nN - \Sigma x_i}{1-p}$$

$$p = \frac{1}{nN} \Sigma_{i=1}^{n} x_i$$

Which is equivalent to:

$$p = \frac{1}{N} E(X)$$

Results

Maximum Likelihood Estimation

The MLE of the parameter p of the quincunx is 0.4995, with a 95% normal based confidence interval of [0.4639,0.5351] calculated as derived above.

Conclusion

Maximum Likelihood Estimation from the 100 samples from the model of the quincunx has estimated the parameter for the binomial distribution to be in the range [0.4639,0.5351]. This would seem to verify that, in fact, even though the quincunx is a non-linear physical device that, overall, it is remarkably fair with p=0.5 within the 95% normal based confidence interval.

The estimated cumulative distribution function also suggests that the distribution will converge binomially. Thus, we can conclude as $n \to \infty$, it will converge on the standard normal distribution as a consequence of the central limit theorem.

Chapter 9

Testing the average waiting time for the Orbiter Bus Service

J Fenemore
and Y Wang
Oct 14, 2007

Abstract

The Metro-owned and oprated Orbiter bus service in Christchurch city is a very popular service that links up some of Christchurch's main suburbs, places and attractions. The timetable provided by the Metro bus company claims that on weekdays between 6 a.m. and 7 p.m., a service will arrive at any given stop every ten minutes, regardless of whether that service travels clockwise or anticlockwise. I hypothesise that this is not the case and that arrivals are influenced by many other factors including current traffic volume, traffic accidents, pedestrian volume, traffic light stoppages and passenger boarding times. We tested this hypothesis by sitting at the UCSA bus stops and recording arrival times.

Motivation

The Orbiter is a highly used bus service and I myself often use this service. Many times while waiting for the service, I have noticed that more often than not, two Orbiter buses arrive at the stop at the same time or within a very short time of each other. Because of logistical reasons, I believe the Metro bus company would not run more buses than needed, meaning that if two buses arrived 'back to back' then there would be a twenty minute wait for the next bus (as the waiting time should be only ten minutes, so for two buses, the time is doubled.) This type of scenario significantly affects the times specified by Metro. For this reason, I believe that in reality, the average waiting time/arrival time is not ten minutes. It is important to note that the timetables distributed by Metro give specific times when buses arrive. These times are all ten minutes apart, which I feel can only be interpreted as meaning that a bus will arrive at a stop every ten minutes and the maximum waiting time for a passenger is also ten minutes. So for the two buses arriving in the 'back to back' situation, while the average time of arrival is presented as every ten minutes on paper, in reality, the buses do not arrive specifically ten minutes apart as claimed. This circumstance also gives a variation of ten minutes and decreases the probability of actually waiting only ten minutes. I wish to address this issue of the average waiting times of the buses in relation to the timetables provided and the variation in actual arrivals. Therefore, by examining the arrival times of the buses and recording waiting times, it can be examined just how accurate the timetables given are and whether they are based on average times or specific times. These issues affect Metro's reliability and credibility.

Method

The experiment we carried out is relatively simple. We sat at the bus stop outside the UCSA building on Ilam road and recorded the arrival times of each Orbiter bus and then calculated the waiting times between each bus. This was done for both clockwise and anticlockwise directions. The waiting time for the first bus in both directions was taken from the time of our arrival to the stop. After that, the waiting time was calculated as the times between bus arrivals.

A range of times were recorded, which covered an entire working day - 8 a.m. to 5 p.m.. These times were recorded on different days to asses not only the time of day but also different days, so we could see how these differeces affect the times. The different times give a fairer assessment of the waiting times. It was assumed that for each day of the week, the waiting times for specific times of the day are relatively the same. A sample taken any day at a specific time would represent all days in the week at that time. The experiment was conducted in this manner because of availability and time restrictions.

While we realise that taking more samples would increase accuracy and reliability while also giving a better description of actual events, we felt it impractical to sit at the stop and record times for an entire day for each day of the week.

Statistical Methodology

For the experiment, we modelled the distribution of the inter-arrival times or waiting times of the Orbiter bus service using the exponential distribution. The probability distribution function is as shown below. The distribution is continuous.

$$f(x;\lambda) = \lambda * exp(-\lambda * x)$$

Where: x is the waiting time, and λ is the rate parameter or $1/mean(x)$.

The mean of this distribution is $1/\lambda$ and has variance $1/\lambda^2$.

The exponential distribution was chosen because of its important memory-less property. Each new waiting time for the next bus is completely independent of the past waiting times. Each bus's arrival is assumed to be independent of the last.

For this experiment, I will be testing whether the average waiting time is ten minutes. More formally:
$\mathbf{H}_0$(null hypothesis):$\mu = 10$ minutes
$\mathbf{H}_A$(Alternative hypothesis):$\mu \neq 10$ minutes

To test this hypothesis, we used non-parametric bootstrap methods to estimate λ and obtain a 95% confidence interval for this value. These values will be formed by sampling the data observed with replacement, at equal probabilities, 132 times, of which an average will be taken. The whole process was then repeated 1000 times. An overall average calculated λ will be then transformed into an average waiting time using the formula:

$$\mu = 1/\lambda$$

where μ is the average.

This will then be compared and contrasted against the average waiting time found by generating 132 realisations of waiting times then calculating the average of these, then repeating this process 1000 times. This is a parametric bootstrap based technique. For this, $\lambda = 1/10$ (where $\mu = 10$ minutes and using the formula above.) An overall average will be found along with a 95% confidence interval for this value. By comparing these intervals and mean values, an accurate decision will be made as to whether buses do arrive on average every ten minutes or not.

Probabilities of certain arrival times around ten minutes will be evaluated to show the accuracy of the service.

The `Matlab` code for this process is given in Appendix III.

Results

The raw data is given in Appendix IV.

The average waiting time for the anticlockwise direction = 9.19 mins.

The average waiting time for the clockwise direction = 8.95 mins.

The total average = 9.07 mins.

The minimum waiting time = 0 mins.

The maximum waiting time =28 mins.

There are 66 waiting time samples for each direction.

Notes for the data:

- Some buses waited at the stop for random amounts of time in order to space the buses apart (this was never for long: 1 or 2 minutes). This was not taken into account when recording arrival times.
- School rush traffic (heavier volumes) was present from 3 p.m. to 3.30 p.m. approx.
- Evening commuter rush was present from approx 4.30 p.m. onwards.
- Morning commuter rush was from 8 a.m. to 9.30 p.m. approx.

Observations on the data: The anticlockwise direction tends to be much more consistent, having a closer average to ten minutes and more observed times close to ten minutes.

Discussion

During less busy hours, buses arrive much more regularly.

The results of the code in (Appendix III) are as follows:

Calculated sample $\lambda = 0.1105$. The calculated 95% confidence interval for this value is [0.1078,0.1129]. The claimed lambda is $\lambda = 0.1$. As the calculated sample is within this interval and the claimed λ is below, we can see that the bus arrival is slightly less than claimed (using $\mu = 1/\lambda$).

Using the claimed λ, the randomly obtained mean value for waiting time is $\mu = 9.9842$. The calculated 95% confidence interval for the mean waiting time is [9.2882,10.6237].

I found from the samples that the anticlockwise, clockwise and total mean waiting times are μ =9.19, 8.95 and 9.07, respectively. None of these values is within the interval previously stated.

When the calculated sample λ of 0.1105 was used to produce the mean waiting time and its 95% confidence interval, the following was produced: μ=9.0350 and [8.4957, 9.6137].

It is important to note that it is seen in the graph, from the empirical CDF, that the probability of having short waiting times is high - the probability of waiting 10 minutes or less according to our observed data is 6288. This value is quite high but the probability of waiting 15 minutes or more is 0.1288 which, in reality, is relatively high also. Practically, this means 1 in every 10 times you wait for an Orbiter to arrive, it will take 15 minutes or more to come. This value may be acceptable by Metro and indeed is good, considering so many unknown factors in traffic, but it would surely frustrate passengers being 5 minutes behind time.

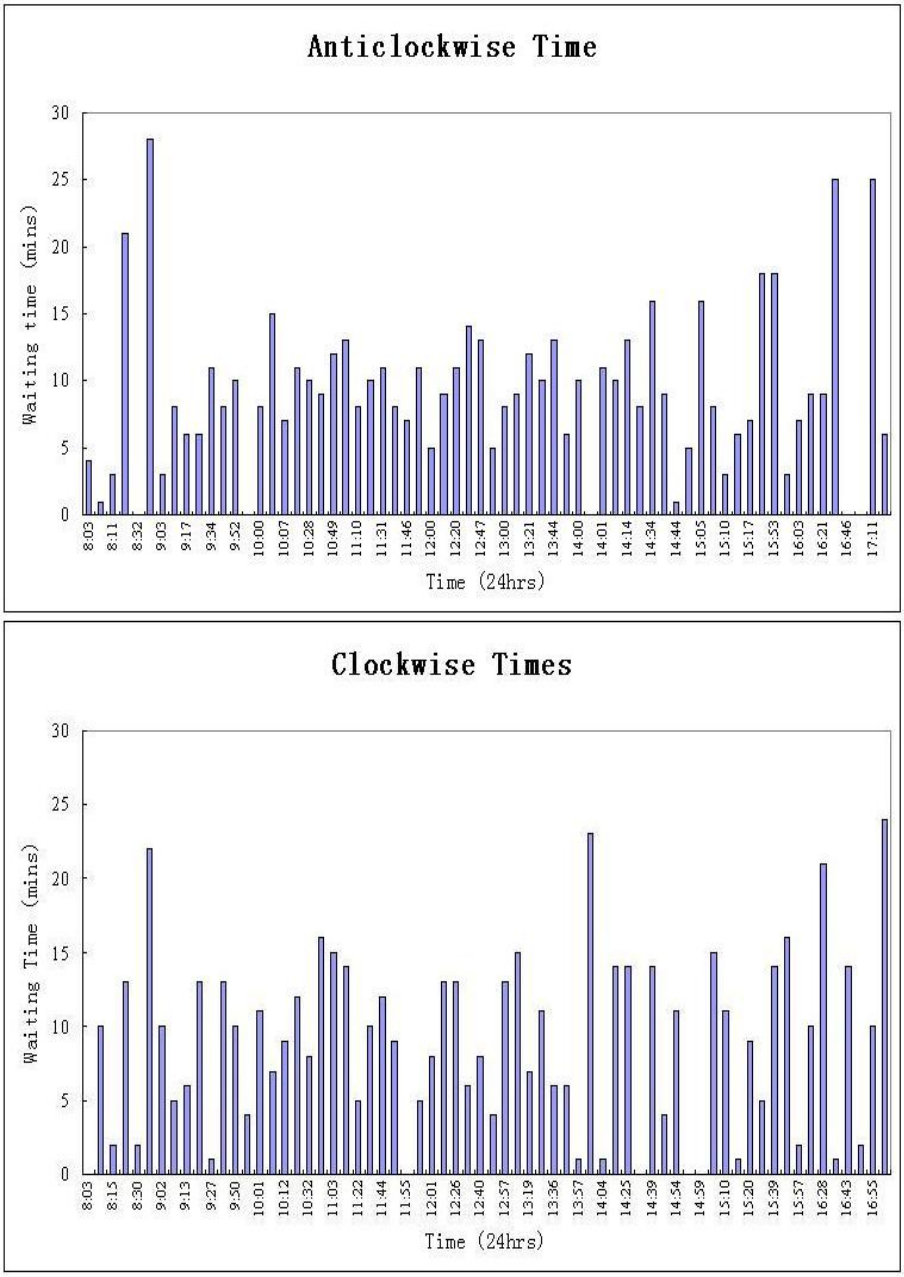


Figure 9.1: From the graphs above, we can see that often, a short wait is followed by a long wait, in both directions. Also, the anticlockwise times are generally much closer to 10 minutes waiting time. It is also seen that around rush hour times (8:30, 15:00, 16:45), a pattern emerged where several buses in quick succession were followed by a long wait for the next bus to arrive. This could be because of the time taken for more passengers than usual to aboard and depart, and areas where traffic volume is greater at these times.

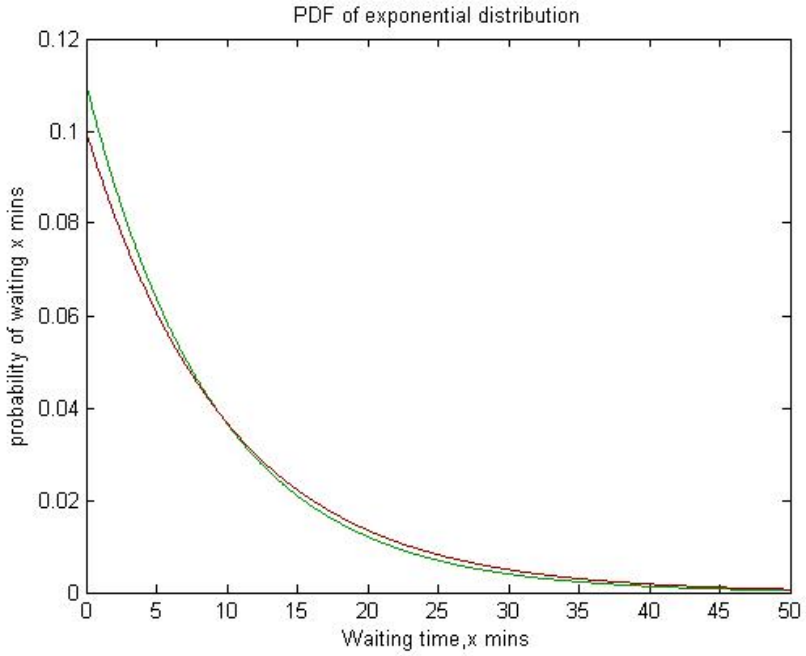


Figure 9.2: This graph shows the probability distribution function for the exponential function with the green line indicating a λ value of 0.1, the claimed λ. The red line indicates the value of λ estimated, 0.1105. From this graph, you can see the probability of getting a short waiting time is high - approximately 0.06, while the probability of a long waiting time is much much lower - approx imately 0.01. The `Matlab` code for this graph is shown in Appendix I.

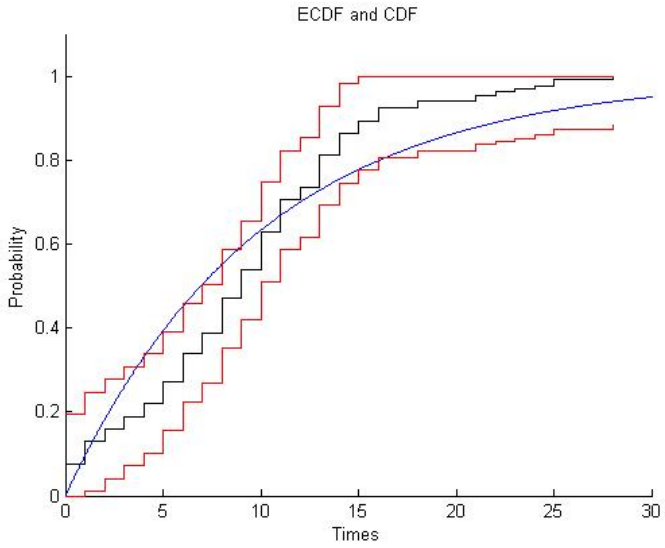


Figure 9.3: This plot is the Empirical CDF plot(black), with a 95% confidence interval (red) and the actual CDF based on claimed $\lambda = 0.1$ (blue). The `Matlab` code for this graph is given in Appendix II. This graph shows the accuracy of the empirical distribution and hence the accuracy of the data we collected. There are some inconsistencies caused by the randomness of inter-arrival times but our empirical CDF is generally good as the actual CDF lies mostly within the interval lines. With more data points, our accuracy would greatly improve.

Conclusion

The calculated value of λ is 0.1105. When this value is used to estimate the mean waiting time, and including the observed waiting times, we can conclude that Metro delivers a service better than claimed. From this λ, we see a mean waiting time of 9.04 minutes - 58 seconds less than the 10 minutes wait claimed.

Furthermore, the mean waiting time estimate calculated and its 95% confidence interval (not including the average waiting times observed) cause us to not accept the null hypothesis, $\mathbf{H}_0$ of $\mu = 10$ minutes at the 95% confidence level.

From all of this, we can confirm that Metro is quite right in claiming an arrival of an Orbiter bus every ten minutes at any stop. In fact, it appears that they do better than this by a whole minute. However, it is all very well to claim this on paper but it is crucial to note that waiting times of 28 minutes do happen, rarely. This illustrates a very important difference between practical and statistical significance. In this case, it has no major effects, as the observed waiting time is less than claimed.

Author Contributions

Josh's Contributions: The original concept; data recordings for Wednesday, Thursday and Friday (5hrs);

data organisation; analysis; methodology and implementation and the preliminary report, final report and presentation notes. *Yirang's Contributions:* Monday's and Tuesday's data recordings (4hrs).

Appendices

I

```
x=linspace(0,50,1000);%Array of x points to evaluate

lambda1=0.1105%Estimated lambda

f1=lambda1*exp(-lambda1.*x);%Calculated probabilities

plot(x,f1,'color',[0 0.6 0])%Plot coloured red
hold

lambda2=0.1%Claimed lambda

f2=lambda2*exp(-lambda2.*x);%Calculated probabilites

plot(x,f2,'color',[0.6 0 0])%Plot coloured green
xlabel('Waiting time,x mins')%Graph titles
ylabel('probability of waiting x mins')
title('PDF of exponential distribution')
```

II

```
antiTimes=[8 3 7 18 18 3 7 9 9 25 0 0 25 6 ...
  10 0 10 8 16 9 1 5 16 6 4 1 3 21 0 28 3 8 ...
  6 6 11 8 10 15 0 8 7 11 10 9 12 13 8 10 11 8 ...
    7 11 5 9 11 14 13 5 8 9 12 10 13 6 11 13];
clockTimes=[0 0 11 1 9 5 14 16 2 10 21 1 14 2 10 ...
    24 6 1 14 14 0 14 4 11 15 0 10 2 13 2 22 ...
    10 5 6 13 1 13 10 11 4 7 9 12 8 16 15 14 5 ...
    10 12 9 8 0 5 13 13 6 8 4 13 15 7 11 6 23 1];
%The raw data-the waiting times for each direction


sampleTimes=[antiTimes clockTimes];%dd all times into 1 array

x=linspace(0,30,1000);%Create array
lambda1=0.1;%Set claimed lambda
f=1-exp(-lambda1*x);%Create cdf realisations based on claimed lambda

[x1 y1]=ECDF2(sampleTimes,7,0,0);
%Call to class distributed ECDF fuction, save output values in arrays x1
%and y1
hold on%Hold plots for superimposition
plot(x,f)

Alpha=0.05;%set alpha to 5%
SampleSize=132;

Epsn=sqrt((1/(2*SampleSize))*log(2/Alpha));%epsilon_n for the confidence band

stairs(x1,max(y1-Epsn,zeros(1,length(y1))),'r');%lower band plot
stairs(x1,min(y1+Epsn,ones(1,length(y1))),'r');%upper band plot
hold off
axis([0,30,0,1])
```

```
title('ECDF and CDF')
xlabel('Times')
ylabel('Probability')
```

III

```
clear
antiTimes=[8 3 7 18 18 3 7 9 9 25 0 0 25 6 ...
  10 0 10 8 16 9 1 5 16 6 4 1 3 21 0 28 3 8 ...
  6 6 11 8 10 15 0 8 7 11 10 9 12 13 8 10 11 8 ...
    7 11 5 9 11 14 13 5 8 9 12 10 13 6 11 13];
clockTimes=[0 0 11 1 9 5 14 16 2 10 21 1 14 2 10 ...
    24 6 1 14 14 0 14 4 11 15 0 10 2 13 2 22 ...
    10 5 6 13 1 13 10 11 4 7 9 12 8 16 15 14 5 ...
    10 12 9 8 0 5 13 13 6 8 4 13 15 7 11 6 23 1];
%The raw data - the waiting times for each direction

rand('twister',489110);%set the seed for rand so results can be reproduced
sampleTimes=[antiTimes clockTimes];%All the sample times collected
lambdaTotal=zeros(1000,1);%An empty array

lambdaObs=1/mean(sampleTimes)%The lambda value for observed samples
lambdaClaimed=1/10 %The lambda claimed by Metro

%This is a non-parametric bootstrap
for j=1:1000%Loop to create 1000 lambdas
    for i=1:132 %A loop to sample with replacement 132 times at equal
        %probability
        u1=rand;%Generate a random number
        x1=deMoivreEqui(u1,132);%Select a random number between 1 and 132
        b(i)=sampleTimes(x1);%Array of random sample times, taken from
        %all samples, using random number generated
    end
    lambdaTotal(j)=1/mean(b);%lambda value for each array of random samples
end

sampleLambda=mean(lambdaTotal)
%The mean lambda for all the lambdas calculted
sortedLambdaTotal=sort(lambdaTotal);%Sort lambdas generated
lowerBound=lambdaTotal(25)%Calculate a 95% confidence interval for lambda
upperBound=lambdaTotal(975)

realisationsClaimed=zeros(1000,1);%An empty array
meanClaimed=zeros(1000,1);%An empty array
%This is parametric bootstrap
for x=1:1000%Loop to create 1000 mean waiting times based on claimed lambda
    for z=1:132 %Loop to generate 1000 waiting times based on claimed lambda
        u2=rand;%Generate a random number
        realisationsClaimed(z)=-(1/lambdaClaimed)*log(u2);
        %Create realisation of x, random number u and lambda claimed
    end
    meanClaimed(x)=mean(realisationsClaimed);
    %Find mean of each array of realisations
end
meanOfClaim=mean(meanClaimed)%Overall mean of realisations created
```

```
meanClaimed=sort(meanClaimed);%Sort array
lowerBound=meanClaimed(25)
%Create a 95% confidence interval for the mean found
upperBound=meanClaimed(975)
```

The above code was wtitten 15/10/07 by J Fenemore and makes use of the follwing function:

```
function x = deMoivreEqui(u,k);
%
% return samples from deMoivre(1/k,1/k,...,1/k) RV X
%
% File Dates : Created  08/06/07  Modified  08/06/07
% Author(s)  : Raaz
%
% Call Syntax:  x = deMoivreEqui(u,k);
%               deMoivreEqui(u,k);
%
% Input      : u = array of uniform random numbers e.g. rand
%              k = number of equi-probabble outcomes of X
% Output     : x = samples from X
%
x = ceil(k * u) ; % ceil(y) is the smallest integer larger than y
% floor is useful when the outcomes are {0,1,...,k-1}
%x = floor(k * u);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

IV. Raw data

Anti-clockwise Route		**Clockwise Route**	
Bus Arrival Times:	Mins waiting:	Bus Arrival Times:	Mins waiting:
15 : 07	8	14 : 59	0
15 : 10	3	14 : 59	0
15 : 17	7	15 : 10	11
15 : 35	18	15 : 11	1
15 : 53	18	15 : 20	9
15 : 56	3	15 : 25	5
16 : 03	7	15 : 39	14
16 : 12	9	15 : 55	16
16 : 21	9	15 : 57	2
16 : 46	25	16 : 07	10
16 : 46	0	16 : 28	21
16 : 46	0	16 : 29	1
17 : 11	25	16 : 43	14
17 : 17	6	16 : 45	2
		16 : 55	10
		17 : 19	24
14 : 00	10	13 : 56	6
14 : 00	0	13 : 57	1
14 : 10	10	14 : 11	14
14 : 18	8	14 : 25	14
14 : 34	16	14 : 25	0
14 : 43	9	14 : 39	14
14 : 44	1	14 : 43	4
14 : 49	5	14 : 54	11
15 : 05	16	15 : 09	15
15 : 11	6		
8 : 03	4	8 : 03	0
8 : 08	1	8 : 13	10
8 : 11	3	8 : 15	2
8 : 32	21	8 : 28	13
8 : 32	0	8 : 30	2
9 : 00	28	8 : 52	22
9 : 03	3	9 : 02	10
9 : 11	8	9 : 07	5

Anti-clockwise Route		**Clockwise Route**	
Bus Arrival Times:	Mins waiting:	Bus Arrival Times:	Mins waiting:
9 : 17	6	9 : 13	6
9 : 23	6	9 : 26	13
9 : 34	11	9 : 27	1
9 : 42	8	9 : 40	13
9 : 52	10	9 : 50	10
10 : 07	15	10 : 01	11
9 : 52	0	9 : 56	4
10 : 00	8	10 : 03	7
10 : 07	7	10 : 12	9
10 : 18	11	10 : 24	12
10 : 28	10	10 : 32	8
10 : 37	9	10 : 48	16
10 : 49	12	11 : 03	15
11 : 02	13	11 : 17	14
11 : 10	8	11 : 22	5
11 : 20	10	11 : 32	10
11 : 31	11	11 : 44	12
11 : 39	8	11 : 53	9
11 : 46	7	12 : 01	8
11 : 57	11		
12 : 00	5	11 : 55	0
12 : 09	9	12 : 00	5
12 : 20	11	12 : 13	13
12 : 34	14	12 : 26	13
12 : 47	13	12 : 32	6
12 : 52	5	12 : 40	8
13 : 00	8	12 : 44	4
13 : 09	9	12 : 57	13
13 : 21	12	13 : 12	15
13 : 31	10	13 : 19	7
13 : 44	13	13 : 30	11
13 : 50	6	13 : 36	6
14 : 01	11	13 : 59	23
14 : 14	13	14 : 04	1