Bayesian Inference for Large Scale Mobile Phone Usage Data

MMathStat Project

School of Mathematics & Statistics
Newcastle University

Sebastian Mellor

May 3, 2012

Abstract

Mobile phones have become an increasing part of our daily lives. There are studies currently investigating *reality mining*, which make use of our mobile data. This project considers the possibility of predicting our next mobile communication using just the data we have about time, location, and call history. A multinomial distribution has been used to model calls and Bayesian information criterion and Bayes factors have been used to implement switching. Results show that day of the week has a significant effect on outgoing calls when used in conjunction with a short list of ordered most likely recipients.

Contents

1. Introduction

1.1. Background

Mobile phones allow us to communicate with our social network while on the move. Technologies and devices such as smart phones encourage us to engage in social activities increasingly often. Social media allows us to communicate in new ways with new people and has become prevalent in business and advertising, as well as enabling communities to share everything and learn together (see Mohr and Paul).

Understanding how we communicate and socialise helps us create better tools for interacting with our digital world. This project considers the possibility of predicting social interactions from real-time data about location and date/time along with previous call history; i.e., using only the last n calls made, can we predict the recipient of the next outgoing call, if told when and where it will be made.

Barzaiq and Loke [2011] discuss the possibility of predicting outgoing calls using the frequency and regularity of calls previously made by the user. They use a fixed equation to weight the importance or monthly, weekly, and daily calls on predicting the next call recipient. We consider this same question of prediction, but create and update a Bayesian model unique to each user considering the factors of current time, date, location, and summary statistics of call history. Our models do not need to store the full records of each individual call to calculate the predictive probabilities nor to update the model.

We perform our analysis on a large data set of calls from Portugal, 2006/7, from which we can select individuals with which to test the accuracy of our models. We have call records for 1.8 million users over the duration of approximately 1 year, this is a total of 435 million call records.

Building our models from this data-set requires an understanding of the data itself and also the problems associated with using such data. The first practical problem is the quantity of data available to analyse. We are interested in designing a suitable model for predicting an individual user's next call, we expect individual users to vary and we must determine which factors are good indicators of calling patterns in general.

A consideration of which we should be aware is that our predictions will likely be performed on mobile devices and although their capabilities continue to improve rapidly we should still consider the memory and computational limitations of these devices. A phone should remain responsive to the users and our predictions should not cause excessive battery drain. Efficient computations are a key factor for these two concerns.

In this project we use several software applications and data resources, see section A.1 for a description of their uses in this project. Several packages and tools are part of a larger area of study called geographical information science or geospatial information studies (GIS).

The next chapter looks at the issues of dealing with large data and the specific information in our file. We then attempt to model the data, with particular reference to an individual handset's call patterns as described in chapter 3. Chapter 4 outlines the methods chosen to simulate the data in order to have an accurate and reliable way of testing our various prediction algorithms, with our general results and conclusions in chapter 5. We briefly discuss cluster analysis and further questions in chapter 6.

2. Large Data

2.1. Introduction

The data set consists of anonymised data of 435 million mobile phone calls in Portugal between 2006 and mid-2007. This covers about 1.8 million individuals which is a significant proportion of the entire population of Portugal which is around 10.6 million (data from WorldBank). We use the term "*individual*" to refer to a single mobile phone number. Some people may have more than one phone number, but the available data does not provide this level of detail.

There are several important details we do not know about the calls or the individuals, including:

- whether more than one Id. belongs to, or is used by, the same person;
- whether the phone is used for business or personal calls;
- name, address, or even phone number of any individual;
- true importance of any location - is it home, work, or something else[1];
- true social ties, relatives or friends; and
- age, gender, or any other personal detail.

The data as provided includes the time of initiation of each call to the second, but this is stored within the date field, making statistics of either date or time alone difficult without first pre-processing; also, of the originating and terminating Id.s there are 1.7 million in the intersection, which means that if desired we could determine return calls and thus analyse continued social interactions between two of the individuals. In this project we do not consider the factor of returned calls although it could be used to filter out certain individuals (e.g., telemarketers), and may provide an improvement to predictions when considering possible missed calls, where missed calls are defined by very short incoming calls.

The key features of the data set is listed in Table 2.1 and a summary of the important values is given in Table 2.2.

[1] Farrahi and Gatica-perez [2009] and Phithakkitnukoon and Horanont [2010] among others attempt to determine the reasons for being near any given cell tower and identify important places.

Mobile Phone Calls	**Cell Towers**
435 million observations	Over 2,000 cell towers
27GB on disk	Cell tower Id.
Caller/recipient (1.8 million unique Id.s)	Longitude and latitude
Originating cell tower	Population of surrounding area
Date and Time	
Call duration (seconds)	

Table 2.1.: Key features of the data.

2.2. Large data

Most of our data is stored in a single large file. The size of this file is around 27GB - much larger than many standard datasets. Due to the size of the data file, we need to use specialised tools.

2.2.1. Challenges

When working with data with *R* the data is unpacked and loaded into internal memory. This is required for the application to perform tasks on the data in a responsive manner, as reading (and seeking[2]) on a mechanical disk is very slow when compared to the speed at which the data is then processed on the CPU or GPU. This causes a noticeable bottleneck when data is not pre-loaded into RAM.[3]

In our case the raw data is already unpacked (and probably takes up less space in memory than as the plain-text CSV file) but is too big to fit in the available memory of an average computer with only 4GB of RAM. This amount of data is too large even for some of the more specialised systems without using a different technique to access and manage it.

To get an idea of the problems we may encounter while working with large data we can consider a few common scenarios:

Scenario 1: Summary Statistics A simple function, looping over the data to determine some summary statistics, such as largest value, smallest value, number of observations, mean, median. These tasks are quick to perform in memory and there are often shortcuts such as computing the number of rows in a table by dividing the total memory consumed by the table by the size allocated to a single row. This requires a strict structure for the table.

[2] *seeking* describes the action when a disk has to move to a specific location in a file to read it.

[3] *RAM* is the high speed memory inside the computer.

	Range				
	Min	Max	Unique Values	Mode	(Frequency)
Caller Id.	226	32177215	1816333	22747234	(117359)
Recipient Id.	22727	32177215	1835148	26021314	(87076)
Cell Tower Id.	3	65441	2384	255	(1619231)
Day Index	91	545	405	327	(1546721)
Duration	1	14865	11295	17	(5138263)
Tower Id.	111	65441	2247		-
Longitude	-28.630	-6.269	-		-
Latitude	32.63	42.13	-		-
Population	0.00	7328.00	-		-

Table 2.2.: Summary statistics for the raw data files. Values computed by aggregating total number of calls by each row's unique values individually.

Scenario 2: Sampling Data Essentially loop over the dataset and select rows matching certain criteria (i.e., passing a certain test). The selected rows must also be stored in memory as a new data set. More complex sampling may require significantly more processing power when applied to a particularly large data-set.

Scenario 3: Aggregating Data Aggregating data is generally more difficult as this combines selecting certain rows and grouping them into a temporary data-set followed by performing summary statistics on each new data-set (each group) before producing the final set of results. This can be efficient when only requiring statistics, such as counting or summing, that can be updated without storing all relevant rows.

Scenario 4: Cross-tabulating Cross-tabulating can consume more memory than aggregation while the processing required would be similar to simple aggregation where counting the frequency for each group is all that is required. The output of a cross-tabulation will be a matrix of size

$$\#\,\text{unique values in column A} \times \#\,\text{unique values in column B}.$$

This matrix may be sparse (many zeros, not too much memory consumed) but if stored as a full matrix then this will probably not fit into memory.

There are several techniques to manage large data sets such as:

- using a powerful machine with enough high speed memory for the data (and resulting statistics);
- using clusters of machines sharing the task by splitting the data and aggregating the results - this is often referred to as distributed computing;

- accessing a dedicated server (with lots of memory) providing highly indexed[4] access to the data for another machine to perform smaller computations with - such a server may provide simple functions such as summations, counting rows, and aggregation; or
- optimising, indexing, or splitting the data files to use locally without reading the entire data files for each use.

A combination of these tools may be used for different aspects of the computations while the actual techniques used, or methods applied to each, are often transferable.

2.2.2. Software optimisations

The method we initially used was combining a reasonably powerful machine with locally splitting the files, since we did not have a dedicated server, or a machine powerful enough to handle the raw data. We used software (`RevoScaleR`) to perform optimisations on the original CSV data files and store the indexed results as new XDF files. This optimisation process may change the way different types of data are stored, especially numerical data such as integers, decimals, and scientific numbers.

The machine used only needs to be powerful enough to manipulate the data in smaller chunks, with enough time very large data-sets can be processed. This applies a few restrictions on the queries that can actually be perform as all functions must be able to run independently without access to the values of other rows, this does not usually cause a problem though. Each optimisation pass takes approximately 2 hours on our machine (4GB RAM, 2 cores (4 with hyper-threading) at 2.3GHz). These optimisations include indexing the data by row, column, and other factors described in the data - the exact algorithms are protected. All we had to do was accurately describe the data - unfortunately this was not possible for the first pass, requiring a second and third pass once we had a better understanding of the data.

With an efficient means of accessing sections of the data files within R we now need to analyse the data itself. This can be done with the same software algorithms that optimised the file, the general method used is *chunking*, where only a portion of data is loaded at one time, for a simple task 500,000 rows may be loaded and operated on at a time, taking about a second per chunk; for 435 million rows this will require only 870 disk read and write operations - approximately 15 minutes.

Consider now the four scenarios introduced in section 2.2.1:

Scenario 1: Summary Statistics Some of these have already been calculated in order to optimise access to the data. Others can now be done in a similar manner

[4] *indexed* access suggests time has been spent analysing the data (possibly at time of creation) so that it has been organised in many different ways already, you would often access this with a *Structured Query Language*

to clustered computing. Although not parallel, the data can be summarised in smaller chunks, with these chunks are then combined and summarised. A simple summary statistic now takes only 12 minutes, these could also have been calculated by streaming[5][6] the data.

Scenario 2: Sampling Data Sampling has been optimised and may not even require a full pass depending on the method used. Complex sampling may require a full pass and take a similar time to new summary statistic. When sampling based on aggregated values, these must be clearly be calculated first and then the desired rows sampled by matching Id. When loading a large array into the chunking function sampling becomes less efficient as this array may be recreated for each chunk or even row.

Scenario 3: Aggregating Data Aggregating data is still the most difficult, but now possible. The indexed data allows quicker access while grouping rows and the chunks allow the functions to be applied. Aggregation can now take anything from twenty minutes to five hours depending on the complexity of the aggregation function applied. Calculating the total number calls by day of week takes fifty minutes.

Scenario 4: Cross-tabulating As with aggregating, cross-tabulating can be done quite easily with the extra overhead of updating the results in two dimensions which will not be negligible. It is easy to attempt a cross tabulation that is too large to store in memory such as total calls by cell tower and originating Id. If this data was stored in a file while processing then reading and writing to disk would become a problem once again.

With basic summary statistics and simple aggregations providing a good understanding of the data structure we can sample subsets of the data that appear interesting, or representative of the whole, with which to perform further analysis, cross-tabulation, and simulations.

2.2.3. Dedicated hardware

Mid-way into the project we gained access to a machine to use as an SQL server, this machine although twice as powerful (8GB RAM, 4 cores at 2.8GHz) has the greater advantage of only being specifically configured and optimised to perform the task of serving data resulting in greater efficiency.

Use of this machine allowed us to submit multiple jobs, in the form of SQL queries, with the results available as another data source (a table in the database). This method of working proved far more effective and allowed for analysis to be performed on our

[5] *streaming* is equivalent to using the smallest chunks possible.

[6] *streaming* reads data from disk sequentially and continuously, only storing the results and never the actual data which is passed directly through functions by row or character.

Personal Computer	**Dedicated Server**
Limited space, suitable for storing results and working with aggregated data.	Several Terabytes of high speed hard disk storage for large data-sets and long results.
Quick to produce visualisations and easy to generate simple statistics of smaller data sets.	Process management for responsive access to large data files and aggregating results.
Many tools available for optimising and analysing data files.	Simple functions available for producing further tables of transformed data.

Table 2.3.: Advantages of different data storage and access methods.

own computers in a more responsive and reactive manner, with access to previous results permanently stored and available.

An example request such as aggregating the number of calls by user would take only half an hour and the resultant data would be immediately available as another data resource. As SQL statements would usually be prepared and submitted a non-interactive work-flow would allow us to continue working with other results. RevoScaleR can still be used to work with the large data when not connected to the server. The package `RMySQL` allows connecting to the database directly from within R, while use of SSH tunnels and port forwarding provide secure access to the server's network while away from the department.

Table 2.3 lists a few advantages of using a personal computer and a dedicated server.

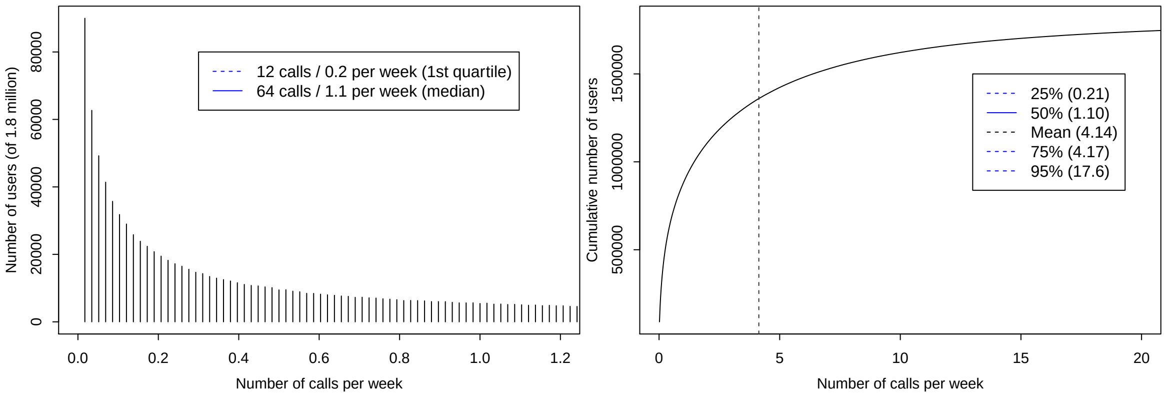
2.3. Visualising the data set

The data was visualised using a combination of R with various GIS packages, and augmenting the images with data from external sources.

Polygons of administrative boundaries and lines representing transport routes have been collected from sources such as DIVA-GIS, these files can usually be read by using the package `maptools`, and often include DBF database files with names, lengths, locations and categories as well as the SHP and SHX shapefiles. Towers within district boundaries can be highlighted by using `point.in.polygon` form the package `sp`.

Other data such as population and altitude is stored in raster files GRI and GRD - this data is of the form of a values on a grid at a specific resolution e.g., 30 arc-seconds and can be read with the package `raster`.

Satellite photography can be downloaded as large multi-resolution files and then manipulated within R, but it was simpler to use the command line tools in the Unified



(a) Number of callers making x calls per week. (b) Cumulative # users making x calls per week.

Figure 2.1.: Graphs showing number of users each making x calls. Data scaled to 'per week' and produced from full data-set (135m calls, 1.8m users).

SDK from LizardTech to convert these files into JPEG images first.

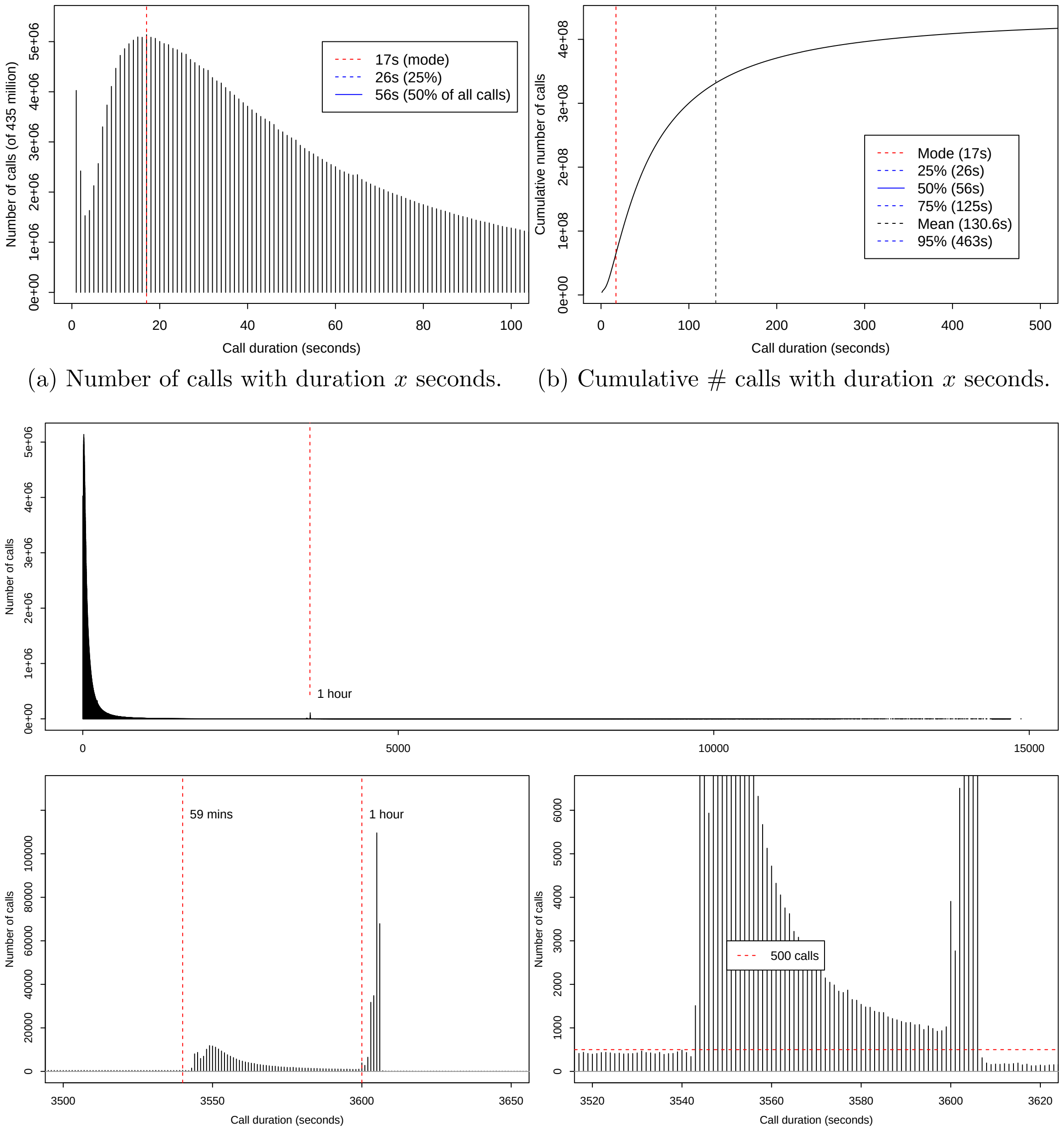
2.3.1. Phone calls

It is difficult to visualise multi-dimensional data for an entire set of phone calls. Due to the quantity of calls - any attempts to visualise every individual call would be infeasible. We have produced summary statistics to determine the range of particular values, the mean, and in some cases the mode. We can now produce histograms of these values to investigate the distribution of calls in more detail.

Callers (Originating Id.)

Looking at the plots in Figure 2.1 we observe that the distribution of calls per user follows an inverse power function/law. We see that the average number of calls made by any user is about 4.1 per week but we see the majority of users making very few calls with 50% making less than 64 calls over the 406 days in the data (1.1 per week) and 95% making less than 1018 calls (17.6 per week, 2.5 per day).

For our investigations we have sampled 1000 users from the top 5% whom are each making at least a thousand calls, we confirm that these users show similar patterns to lower usage users. We also sampled 1000 users from the full data set, and a smaller sample of one hundred high usage user. The figures up to section 2.3.5. are all produced from the full data-set unless otherwise stated.



(a) Number of calls with duration x seconds.

(b) Cumulative # calls with duration x seconds.

(c) 3 graphs displaying a feature at 3600 seconds. Above, full range of duration values. Left, 3 minute range around the feature. Right, y-stretched plot showing drop from before to after the feature.

Figure 2.2.: Call durations for all 135 million calls. Highlighting the initial pattern, overall shape, and a single feature at 3600 seconds.

Call duration

Figure 2.2a gives the call durations. This figure shows that most users make calls of seventeen seconds in length, while many people make very short calls with a significant number of calls less than 3 seconds. The very short calls are probably missed calls, i.e., call answered by an answering machine.

After seventeen seconds the frequency of each call duration slowly declines with 50% of all calls less than 1 minute (56 seconds), 75% less than 125s (approximately two minutes), and 96.6% less than ten minutes. The longest recorded call is 4 hours 7 minutes 45 seconds. Although not many calls are, proportionally, longer than 10 minutes there is a significant peak of calls that are within a few seconds of being exactly 1 hour, with the highest point on this peak having an equivalent volume of calls at the 93.1% limit (6 minutes 9 seconds). In other words, the calls of duration around 1 hour would be included in a multi-modal 95% interval.

Unfortunately, looking at the distributions of call duration for each individual user would produce another huge (1.8m by 11k possible rows) data-set, as this is not directly related to our question we have left several such statistics and investigations as further work. One such question that would be interesting to answer relates to the pattern visible at around 3,600 seconds, or one hour. In Figure 2.2c it can be seen that within only one minute a seemingly smooth, shallow curve suddenly peaks twice leaving the following curve much lower. It would be interesting to see whether all individuals demonstrate the same pattern.

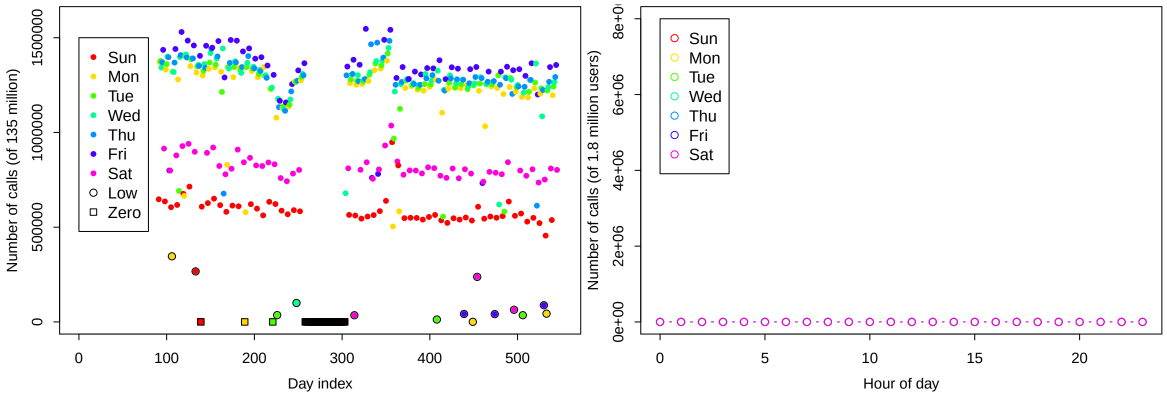
Call date and time

The date and time are combined into a single integer value, i.e.,

$$\text{DateId} = 100,000 \times \text{Days since 2006-01-01} + \text{Seconds this day}.$$

This made it initially difficult to produce any plots of the date/time distribution for the entire data-set. We have since created a full copy of the original data with separate "Day Index" and "Time of Day" fields.

Figure 2.3a shows us some missing data for 3 isolated days, with an extended period of missing data between 2006-09-16 (Sat) and 2006-10-31 (Tue). There are also some days with very few recorded calls, as low as 637 calls on 2007-03-26 (Mon). Aside from the missing data we also notice some distinct bands in call frequency. The day of the week is indicated by colour and it appears that the day of the week has a significant effect on the number of calls being made. No apparent pattern is discerned in terms of day of week affecting the unusually low number of calls otherwise. There seems to be a dip in calls made in mid to late August '06 (233–241), and an increase leading up to 25th December (358) with a significant decrease on the days surrounding the 25th.



(a) Number of calls made each day between 2006-04-02 (Sun, 91) and 2007-06-30 (Sat, 545).

(b) Number of calls made each hour, grouped by day of week. 1.8 million users and 405 days.

Figure 2.3.: Number of calls made on a daily basis.

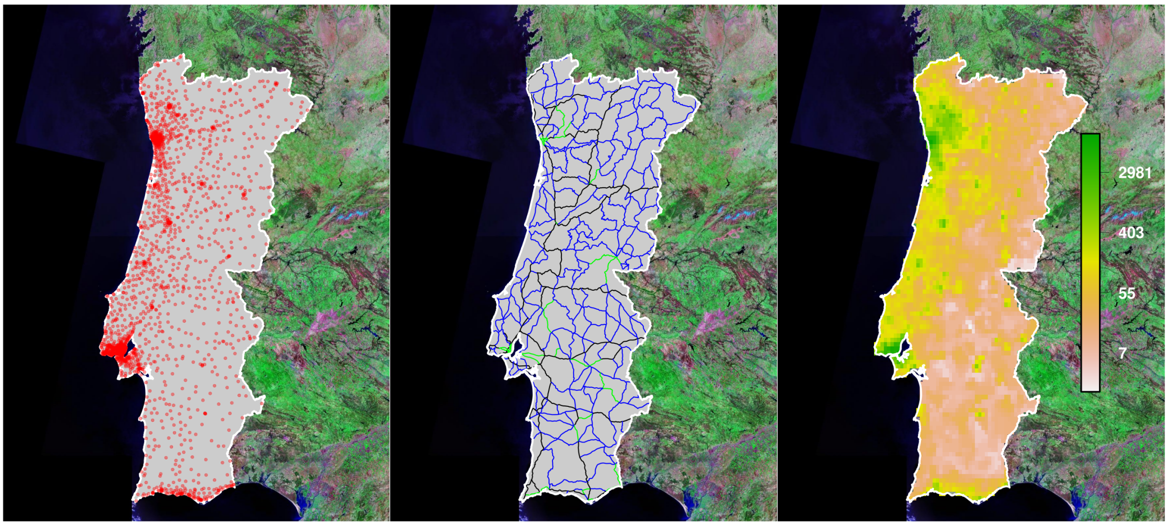
Figure 2.3b took much longer to produce as the data had to be cross tabulated with a modulo function as one of the fields. By using the SQL server this was achieved in 1.5 hours. We produced this plot after seeing the pattern produced with a smaller sample shown in Figure 2.6b. Although our data is discrete, a histogram would not easily allow us to overlay the data for the 7 days of the week. For this reason the points representing number of calls made each second have been connected with a line, and we can now see that there is a clear pattern during the weekdays.

Figure 2.3b shows an increase in calls leading up to 12 o'clock and 5 o'clock in the afternoon, presumably these relate to lunchtime and the end of the usual working day. We also see that Saturday and Sunday display similar increases around lunchtime with a lower over overall call volume.

2.3.2. Cell towers

Figure 2.4 shows the distribution of mobile phone cell towers across the country of Portugal, in Figure 2.4a each tower is represented as a small, translucent, red circle. We can see tendencies for towers to be placed close to transport routes which can be seen in Figure 2.4b, and a much higher density in the larger cities than in areas of lower population density with population density visible in Figure 2.4c.

Figure 2.5 shows a magnified image of the area around Porto, onto which a Voronoi diagram has also been included. A Voronoi diagram breaks up the area into Voronoi cells determined by the distances to surrounding cell towers. All points within a cell must be closer to the contained tower than to any other tower. It is likely that a call made while within each area, or cell, would be connected via the corresponding tower as this would likely have the strongest signal. Thus, we have an approximate resolution for location accuracy for a user - clearly this varies greatly between cell towers and



(a) Mobile phone cell towers. (b) Roads and transport routes. (c) Population density ($/km^2$).

Figure 2.4.: Distribution of cell towers in Portugal and other spatial features

regions, with a possibly higher resolution near the coast in Porto. This can also be misleading as several towers may actually be in range.

2.3.3. Data-set deficiencies

In the data set, there are approximately thirty calls where the originating Id., tower, duration, or date was set as NULL. These points have been discarded before any analysis. Additionally, there are two calls with day indexes of -1 or 1 - as all other calls have index 91 and above we have removed these two as well.

Finally, with a data-set of this size it is not always easy to determine the exact structure of the data, and thus many errors or other anomalies that may be present that will be missed. We performed some specific tests on the data to determine a few important details about the data.

Some such possible errors are:

- no calls made on 2006-05-20 (Sat), 2006-07-09 (Sun), 2006-08-10 (Thu), or between 2006-09-16 (Sat) and 2006-10-31 (Tue);
- unusually low number of total calls on 13 other dates are also visible in Figure 2.3a;
- the distribution of Id. numbers for users and towers seems irregular but we cannot know if that is deliberate or not;
- several towers have been given associated population densities of zero in the

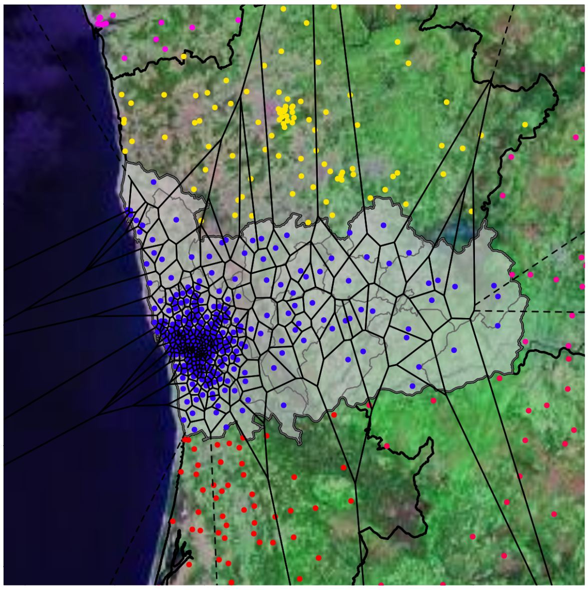


Figure 2.5.: Distribution of cell towers in the district of Porto with a Voronoi mosaic of the towers within Porto.

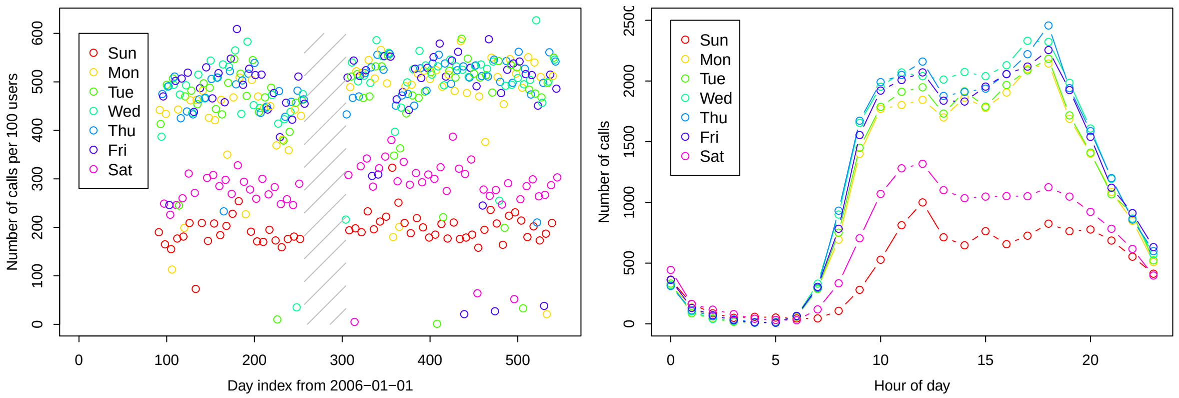
original data set. This measure of population is not used in this project and instead we overlay raster data from DIVA-GIS. This is an example of errors introduced by mismatches when combining multiple data sources;

- some individuals only call one other user, sometimes making a very large number of calls of the order of several hundred; and
- we may consider the age of the data to reduce the relevance of this project to today's mobile phone usage and while today's usage is likely quite different this does not detract from any of the methods or results we develop or discover.

2.3.4. Extending the data

We can use data from other sources to help interpret the data we have, for example, we can use daily or hourly weather reports for Portugal and specific airports over the two year period provided by Weather Underground.

As visible in several of the figures displaying cell tower locations, we have used data for district borders, populations, roads and railways, and municipalities provided by DIVA-GIS. As well as using satellite photography available from NASA.



(a) Calls per day for 100 high usage users.

(b) Number of calls per hour for a sample of 100 high usage users.

Figure 2.6.: Number of calls daily and hourly for a sample of 100 users.

2.3.5. Data for modelling

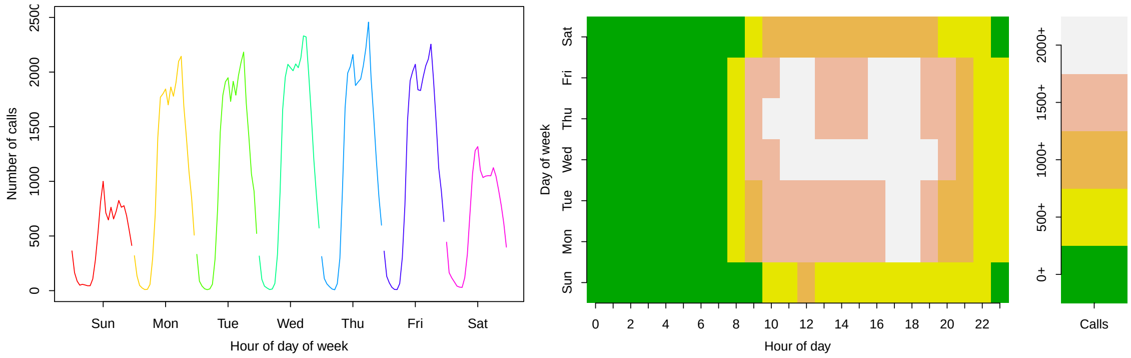
For the remainder of the project we consider a representative sub-set of the data as working with the full set would be infeasible. We are interested in individual users while producing the predictive models, and a sample of one thousand originating call ids (users) should provide us with enough data to test the effectiveness and suitability of each model.

To get a suitable representation of the data we take a random sample of one thousand users from the full data set and another sample of one hundred users from the top top 5% usage range.

Comparing the plots in Figures 2.6 and 2.3 we can see that the distribution of calls has very similar patterns at least with regard to the times of day and days in the year. These plots were produced from the sample of only one hundred users within the top 5%. This indicates that either the usage patterns visible here are equivalent for most users independent of usage level or that the patterns visible are dominated by the high usage users.

We can work more quickly when handling smaller samples of data and this gave us time to investigate a few more representations of the same data such as those shown in Figure 2.7. Figure 2.7a is simply a concatenation of the curves shown in Figure 2.6. From this new plot we can clearly see the repeating daily pattern of mobile phone usage, as well as the reversed peaks during the weekend.

Figure 2.7b is an image plot, or level plot, of the same data using colour to represent the volume in each hourly block. We can see suggested similarities in the call volumes around 17:00 on weekdays, 11:00-19:00 on Wednesday, and 11:00 on Thurday and Friday. Call volumes on Sunday also appear to be similar to call volumes in the later evenings on th eother days. The breaks chosen to distinguish each level can significantly



(a) Number of calls made per hour for the duration of one week.

(b) Image plot of calls per hour

Figure 2.7.: Daily call volumes

change the appearance of this image. With breaks of 5000 we can get at least a basic suggestion of similarity in call volumes.

With samples of one thousand users from the full data-set and one hundred higher usage users we can select a small number to help develop our initial models and hence determine whether it is possible to predict most likely next outgoing call.

3. The Models

3.1. Introduction

We are analysing individual call patterns and attempting to predict the next call. To get reasonable results at this stage it is helpful to sample a small number of users with a higher mobile phone usage. We do not feel that this will restrict our models as the current trend is towards higher usage mobile phone users (see Pettey [2011]).

We have shown in the previous chapter (see Figures 2.6) that a comparatively small sub-set of users still exhibits similar population statistics to the whole data-set. We can quickly analyse this for the important factors in our models and choose another smaller number of users such that they represent the full range of variability shown..

3.2. Independence model

We can consider the probability of calling any recipient as constant value, not affected by any previous calls made to the same or other recipient, nor by any external conditions such as time. Each call in this model will be independent.

Our independence model will be used as a basis for further models and will be used to qualitatively determine whether any improvements are being made.

We describe the call history for a single individual as a sequence of observations, $y_1, y_2, \dots, y_n$, of the random variables $Y_1, Y_2, \dots, Y_n$ where Y_t is the terminating Id. at position t in time order. Y_t has state space $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_l\}$, with $\mathcal{C}$ being the set of possible terminating ids (recipients).

We start out with a simple model where Y_t are independent and that we have

$$Pr(Y_t = \mathcal{C}_1) = p_1, \quad Pr(Y_t = \mathcal{C}_2) = p_2, \quad \dots, \quad Pr(Y_t = \mathcal{C}_l) = p_l \tag{3.1}$$

with $\sum_{j=1}^{l} p_j = 1$, and $p_j > 0$.

We fit this model to a given call history using the maximum likelihood to estimate the probabilities $\boldsymbol{p} = (p_1, p_2, \dots, p_l)^T$. In general it can be shown that when a call to $\mathcal{C}_1$ has occurred n_1 times, a call to $\mathcal{C}_2$ has occurred n_2 times, and so on, then the

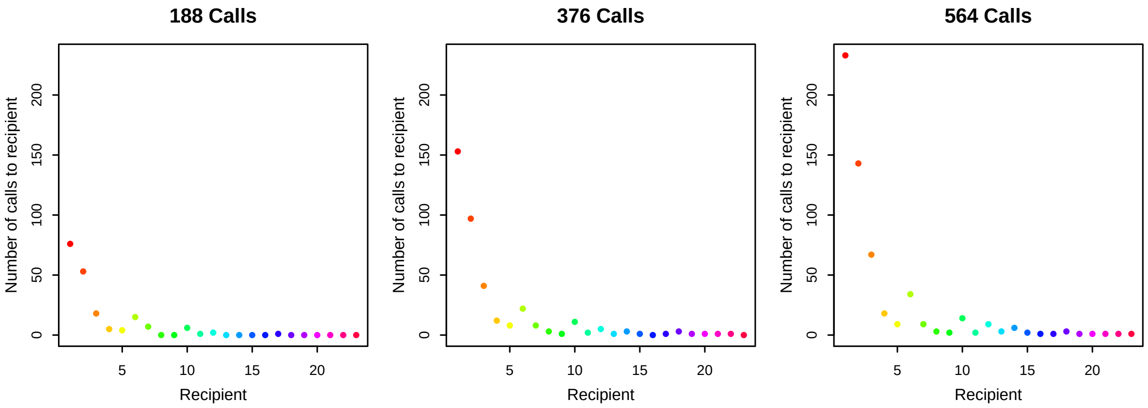


Figure 3.1.: Number of calls made to each of 23 recipients after 188, 376, and 564 total calls for user with Id. 23352025.

likelihood function is

$$L(\boldsymbol{p}|\boldsymbol{y}) = p_1^{n_1} p_2^{n_2} \cdots p_l^{n_l} = \prod_{j=1}^{l} p_j^{n_j}. \tag{3.2}$$

The maximum likelihood estimate for the probability p_j is

$$\hat{p_j} = \frac{n_j}{n} \tag{3.3}$$

where n is the total number of calls made.

Figure 3.1 shows that the probability of calling a recipient tends to a fixed probability mass function as apparent by the visible shape becoming stronger in each plot.

Aside: An alternative model was investigated where the number of calls was additionally weighted by the time since each call, this allowed the model to adapt to changes in call pattern. Further investigation showed that this was not necessary as call patterns do not change significantly over the two year period.

3.3. Dirichlet distribution

Since our model requires constant parameter updates with the arrival of each new call, we will use the Bayesian paradigm. The conjugate prior for the multinomial distribution is the Dirichlet density (see Gu for some more information)

$$(p_1, p_2, \ldots, p_l) \sim \mathfrak{D}(a_1, a_2, \ldots, a_l), \quad \text{with } a_i > 0. \tag{3.4}$$

This is a continuous distribution with probability density function (pdf)

$$\pi(p_1, p_2, \ldots, p_l) = k p_1^{a_1 - 1} p_2^{a_2 - 1} \cdots p_l^{a_l - 1} \tag{3.5}$$

where k is chosen to ensure the pdf is valid. We choose a suitable prior by specifying values $\boldsymbol{a} = (a_1, a_2, \dots, a_l)$.

Suppose $a_j = bm_j$, with $m_1 + m_2 + \ldots + m_l = 1$, then

$$E(p_j) = \frac{a_j}{a_1 + \cdots + a_m} = \frac{m_j}{m_1 + \cdots + m_l} = m_j \tag{3.6}$$

$$\begin{aligned} Var(p_j) &= \frac{a_j(a_1 + \cdots + a_l - a_j)}{(a_1 + \cdots + a_l)^2(a_1 + \cdots + a_l + 1)} \\ &= \frac{m_j(1 - m_j)}{b + 1} \end{aligned} \tag{3.7}$$

clearly we can maximise variance by choosing a value for a that is small. We can also choose values for $\boldsymbol{a}$ such that the prior has constant density $\pi(\boldsymbol{p}) = k$ i.e., $\boldsymbol{a} = (1, 1, \dots, 1)$, $b = 1/l$.

Our posterior beliefs can now be shown to be

$$\begin{aligned} \pi(p_1, p_2, \dots, p_l) &\propto \pi(p_1, p_2, \cdots, p_l) \times L(p_1, p_2, \cdots, p_l | \boldsymbol{y}) \\ &\propto p_1^{a_1+n_1-1} p_2^{a_2+n_2-1} \dots p_l^{a_l+n_l-1} \end{aligned} \tag{3.8}$$

and thus

$$(p_1, p_2, \dots, p_l)|\boldsymbol{y} \sim \mathfrak{D}(a_1 + n_1, a_2 + n_2, \dots, a_l + n_l). \tag{3.9}$$

We can now show that the mean and variance of p_j are as follows:

$$E(p_j|\boldsymbol{y}) = \frac{b}{b+n} m_j + \frac{n}{b+n} \hat{p_j} \tag{3.10}$$

$$Var(p_j|\boldsymbol{y}) = \frac{E(p_j|\boldsymbol{y})\{1 - E(p_j|\boldsymbol{y})\}}{b + n + 1} \tag{3.11}$$

with $\hat{p_j}$ as in Equation (3.3).

3.4. Conditional model

The independence model we have just described could be compared with a simple list of likely recipients, ordered by total calls which is indeed the main driving factor of the model. We wish to improve our predictive model by having conditional probabilities.

A common conditional model used in other situations would be a 1st Order Markov Chain, in our case this is equivalent to predicting the next outgoing call based on the previous outgoing call. Initial investigations conclude that this model is of little practical use. Instead we will consider a conditional model using date/time information to determine whether we have a weekend or a weekday call and predict accordingly.

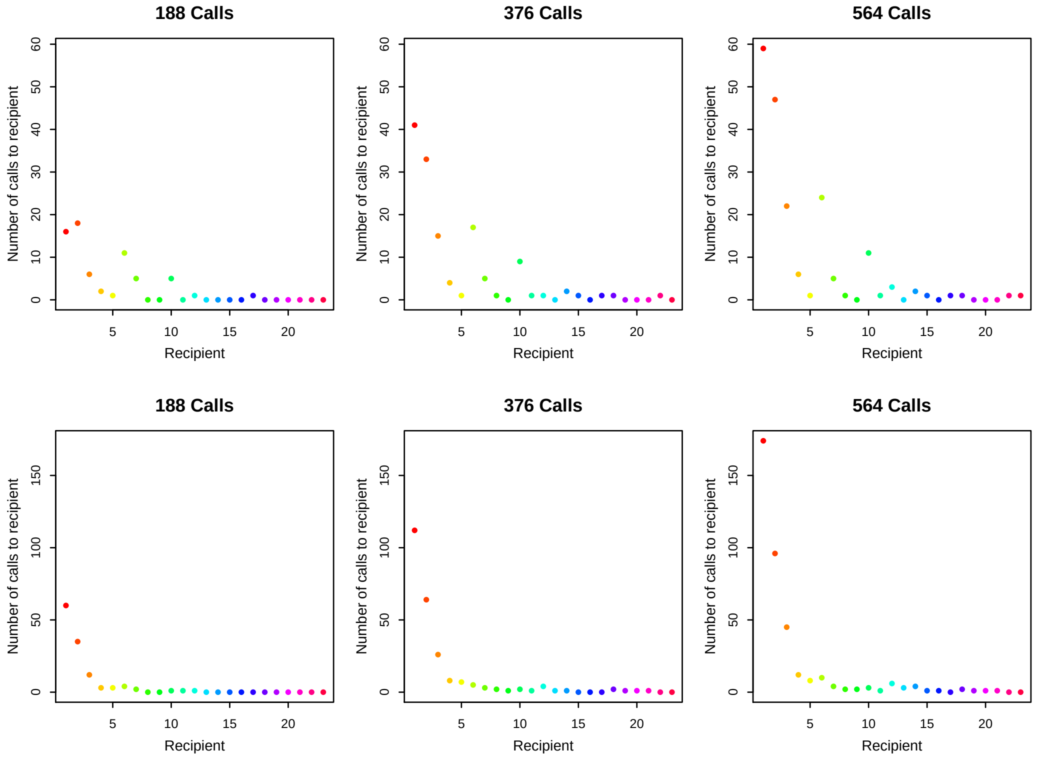


Figure 3.2.: Number of calls made to each of 23 recipients (with condition) after 188, 376, and 564 total calls for user with Id. 23352025. Top row: weekend calls only. Bottom row: weekday calls only.

We can now define our model such that for $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2\}$ representing weekday or weekend,

$$Pr(Y_t = \mathcal{C}_j|\mathcal{D}_1) = p_{1j}, \quad Pr(Y_t = \mathcal{C}_j|\mathcal{D}_2) = p_{2j}, \tag{3.12}$$

and we can produce a probability matrix

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1l} \\ p_{21} & p_{22} & \cdots & p_{2l} \end{bmatrix}. \tag{3.13}$$

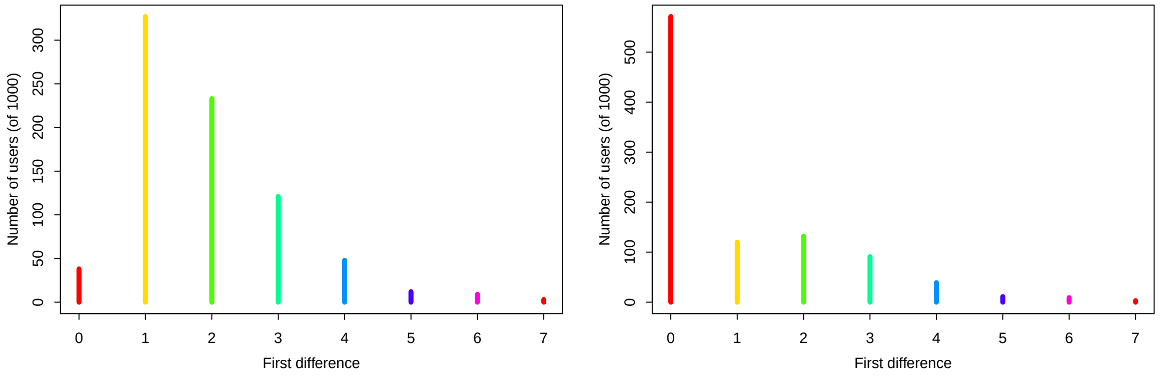
The likelihood function for this model is

$$L(\boldsymbol{p}|\boldsymbol{y}) \propto \prod_{i\in\{1,2\},j\in\{1,\dots,l\}} p_{ij}^{n^{ij}} \tag{3.14}$$

with $n_{ij} = \sum_{i\in\{1,2\},j\in\{1,\dots,l\},t\in\{1,\dots,T\}} \mathbb{I}(Y_t = \mathcal{C}_j|\mathcal{D}_i)$

$$\hat{p_{ij}} = \frac{n_{ij}}{n_i} \tag{3.15}$$

where $n_i = \sum_{j\in\{1,\dots,l\}} n_{ij}$.



(a) With a sample size of 1000 users. Zero indicates users who made only one call.

(b) With a sample of 1000 users with at least 100 calls. Zero indicates users who made less than 100 calls.

Figure 3.3.: Number of users with a difference in ordered weekend/weekday lists at position x.

Figure 3.2 shows the call distriubution split by weekend/weekday. On weekends this user appears to call recipients 6 and 10 significantly more than during the week. If we consider estimating the top 5 most likely recipients to be called we would clearly choose numbers 1, 2, 6, 3, and 10 in that order if it was a weekend, and 1, 2, 3, 4, and 6 if not.

It would seem that the most likely recipient will probably be constant even if we add more conditional factors We can determine how many of the top recipients are constant for a given factor such as weekend/weekday and use that to investigate the importance of such factors in predicting phone calls for the larger sample of one thousand.

Figure 3.3 shows the number of users for whom a weekend/weekday model would make a difference and the position of the first value that would be different. We have 38 users who made only 1 call. There are 327 users for whom the weekend/weekday model would make a difference when used to predict the next call exactly, and 233 for whom weekend/weekday effects who is likely to be called in an ordered list of length 2.

These results are surprising as when looking at individual users who make a significant number of calls we do not see such a variation in the top two. If we only consider users who make at least 100 calls then we exclude just over half of the users and now see that the list must contain the top two likely recipients to make the greatest improvement. It is clear that a user making more calls may regularly call more people, but these results show that a significant improvement can still be made by considering weekend or weekday as a factor in the predictive model for next call of most likely next calls.

4. Simulating Data

4.1. Introduction

Our aim is to be able to predict the next outgoing call any user might make at a given time and we will be doing this by choosing the most likely outcomes from a dynamically updated model based on the models in chapter 3, or *on-line learning*.

We wish to determine how many calls are required before a predictive model will reach it's optimum fit, and thus the best predictions it can make. This value may be larger than the number of calls available for a particular user, or we may wish to test the limits of the predictive model. We can achieve this by creating a model to simulate random calls from a particular user's full call history.

4.2. Simulation

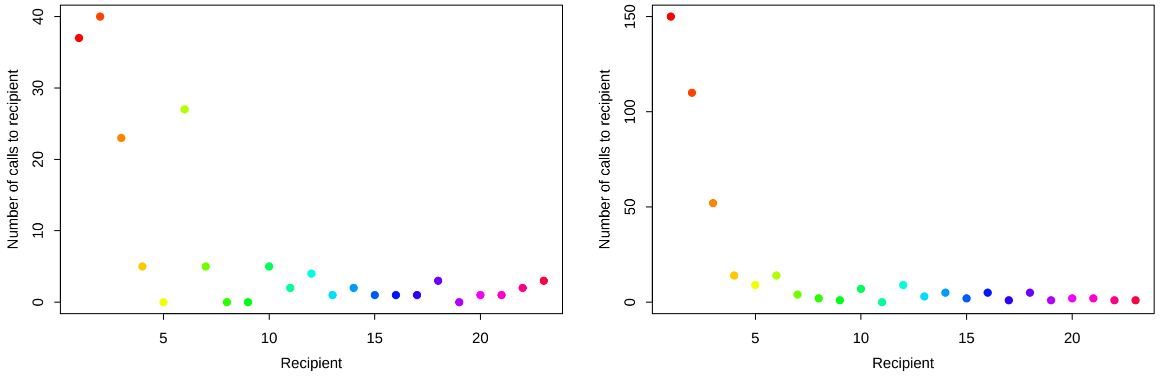
Simulating data for this process is straigthforward:

1. Initialise model parameters;
2. Probability of weekend call is *weekend calls/total calls*;
3. With probability from (2), simulate a call using the multinomial distribution for a weekday call and otherwise simulate a call for a weekend call.

The simulated data is shown in Figure 4.1. Simulation parameters were chosen so that the simulation data qualitiviely matches the data in Figure 3.2. Recipients 6 and 10 are again favoured at weekends only.

We can now run repeated simulations for a particular user's usage pattern and determine how well each model performs at predicting these calls.

By simulating our own data we have control over the parameters used in the generating model, allowing us to have a more reliable comparison of the performance of the predictive models to the 'true' structure. Simulated random data also allows us to run predictive algorithms for much longer as well as repeatedly for the same individual's call patterns giving us a better range of data for producing visualisations.



(a) Number of calls to each recipient during the weekend for simulated data.

(b) Number of calls to each recipient on a week day for simulated data.

Figure 4.1.: Number of calls to each recipient from simulated data set.

4.3. Prediction

Figure 4.2a shows the cumulative success rate of both the independent predictive model and the weekend conditional predictive model. The plot is very noisy when only a few calls have been made - one correct call after only 5 total will increase the success rate by 0.2. We see the conditional model improving more quickly than the independent model in this particular simulation.

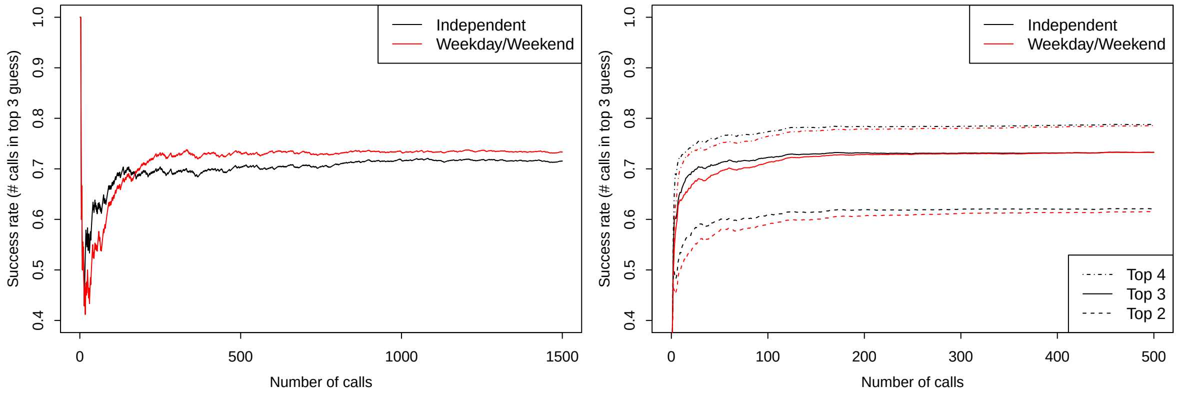
When the simulation is run 100 times as in Figure 4.2b and the average success rate taken we see that the two predictive models become closer in quality. The improvement gained by increasing the list size is far greater than that gained by the conditional model alone. It can be seen that the conditional model generally performs slightly worse than the independence model for the first couple of hundred calls. This is due to the extra parameters required to build the model. The exact point where the conditional model surpasses the independence model varies greatly between individuals and also with list size - if this happens at all.

We clearly need a suitable method of choosing the model that is performing most effectively at any given time.

4.4. Model switching

4.4.1. Bayesian Information Criterion

With more than one model available to us we wish to determine which performs better so that we may use it. We could not do this manually offline as there is not likely to be a single model that fits well all of the time or for every individual.



(a) Cumulative success rate for next call being in top 3 predicted recipients.

(b) 100 averaged simulations showing different predictive list lengths.

Figure 4.2.: Success rate. Number of calls within the predictive list for each model and varying list sizes.

To do this we can look at the value of the Bayesian Information Criterion (BIC, or *Schwarz information criterion*). This is similar to the Akaike Information Criterion (AIC), a measure of goodness of fit often used in linear regression, although more weight is given to penalising the number of parameters in the model when using the BIC. This will help us as we expect more complex models to take longer before they become as efficient as more basic equivalent models.

AIC and BIC are defined as follows:

$$\mathrm{AIC} = -2\log L(\hat{\boldsymbol{p}}|\boldsymbol{y}) + 2k \tag{4.1}$$

$$\mathrm{BIC} = -2\log L(\hat{\boldsymbol{p}}|\boldsymbol{y}) + k\log_e n \tag{4.2}$$

where L denotes the likelihood function for each model, k is the number of parameters in the model, and n is the number of observations so far.

We can select the model corresponding to the lowest BIC value at each iteration and thus select the model with the highest Bayesian posterior probability. We can derive the BIC with the aid of work done by Cavanaugh [2009] and Kass and Raftery [1995].

Derivation

Let $\boldsymbol{y}$ denote the observed data (i.e., each call being made and current weekend/weekday status), and let M_k be our candidate models ($k = 1, 2$) with a unique selection of some parameters $\boldsymbol{\theta}_k$ from the parameter space $p_1, \ldots, p_l, p_{11}, \ldots, p_{2l}$. We now have the likelihood of each model $L(\boldsymbol{\theta}_k|\boldsymbol{y})$ and the maximum likelihood estimate $\hat{\boldsymbol{\theta}}_k$.

Let $\pi(k)$ denote a discrete prior for the model M_k, and $g(\boldsymbol{\theta}_k|k)$ a prior on $\boldsymbol{\theta}_k$ given the model M_k. Now with Bayes' theorem the joint posterior of M_k and $\boldsymbol{\theta}_k$ is

$$h((k, \boldsymbol{\theta}_k)|\boldsymbol{y}) = \frac{\pi(k)g(\boldsymbol{\theta}_k|k)L(\boldsymbol{\theta}_k|\boldsymbol{y})}{m(\boldsymbol{y})}$$

with $m(\boldsymbol{y})$ the marginal distribution of $\boldsymbol{y}$.

Thus, the posterior probability for M_k is given by

$$P(k|\boldsymbol{y}) = m(\boldsymbol{y})^{-1}\pi(k)\int L(\boldsymbol{\theta}_k|\boldsymbol{y})g(\boldsymbol{\theta}_k|k)d\boldsymbol{\theta}_k.$$

If we minimize $-2\log P(k|\boldsymbol{y})$ we have

$$-2\log P(k|\boldsymbol{y}) = \cancel{2\log m(\boldsymbol{y})} - 2\log \pi(k) - 2\log\left\{\int L(\boldsymbol{\theta}_k|\boldsymbol{y})g(\boldsymbol{\theta}_k|k)d\boldsymbol{\theta}_k\right\}$$

with the first term being constant with respect to k.

We can obtain an approximation of the integral above by using a second-order Taylor series expansion about $\boldsymbol{\theta}_k$.

$$\begin{aligned}
-2\log P(k|\boldsymbol{y}) &= -2\log\pi(k) - 2\log\left\{\int L(\boldsymbol{\theta}_k|\boldsymbol{y})g(\boldsymbol{\theta}_k|k)d\boldsymbol{\theta}_k\right\} \\
&= -2\log\pi(k) - 2\log\left[L(\hat{\boldsymbol{\theta}}_k|\boldsymbol{y})\left(\frac{2\pi}{n}\right)^{k/2}|\bar{I}(\hat{\boldsymbol{\theta}}_k, \boldsymbol{y})|^{1/2}\right] \\
&= -2\log\pi(k) - 2\log L(\hat{\boldsymbol{\theta}}_k|\boldsymbol{y}) + k\log\left(\frac{n}{2\pi}\right) + \log|\bar{I}(\hat{\boldsymbol{\theta}}_k, \boldsymbol{y})|^{1/2} \\
&\approx -2\log L(\hat{\boldsymbol{\theta}}_k|\boldsymbol{y}) + k\log n
\end{aligned}$$

where we are ignoring terms that are bounded as $n \to \infty$.

Model switching with the BIC

We can see in Figure 4.3 that the BIC is a good indicator of the best model and in the case shown ultimately performs better than either the independent model or the conditional model alone. The BIC favours the independence model initially after very briefly (3 or 4 calls) favouring the conditional model. As the conditional model accumulates enough data we can see that the probability of switching back to this model for further predictions increase steadily and thus keeping the success rate up.

We can see clearly the effect of these choices as the initial choice leaves success rates just below the independence model up to 350 calls. When the BIC passes 0.5 the dynamically switching model surpassing the independence model by using the predictions from the conditional model. This result is not always seen clearly, with the

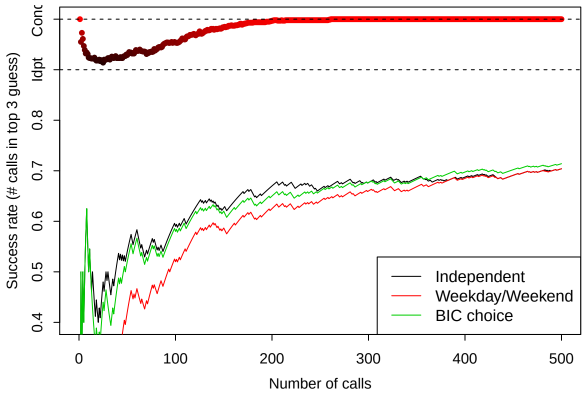


Figure 4.3.: 100 averaged simulations with the BIC used to select model. Average BIC value and desicion is indicated above the success rates between Idpt–Cond. C.f. Fig 4.2b

BIC switching model occasionally slipping below both other models, but in general it will stay just above the midpoint of the two choices.

We may wish to calculate the BIC for several models for each prediction. Certainly in the simulations the computational overhead of 100,000 likelihood calculations (500 calls × 100 simulations × 2 models) can take some time. We may wish to run these predictions on a mobile phone in real-time and any optimisations we can make will be helpful; considering the Bayes factors may help us with this.

4.4.2. Bayes factors

The Bayesian information criterion can be considered an approximation to the *Bayes factors*. The calculations for Bayes factors often include analytically and numerically difficult integrals and so the BIC is chosen instead.

In the case of multivariate data and a Dirichlet conjugate prior we have a posterior density that we can integrate using standard techniques, and thus the Bayes factor can be calculated as:

$$B_{12} = \frac{Pr(\boldsymbol{D}|M_1)}{Pr(\boldsymbol{D}|M_2)} = \frac{\int Pr(\boldsymbol{\theta}_1|M_1)Pr(\boldsymbol{D}|\boldsymbol{\theta}_1, M_1)\, d\boldsymbol{\theta}_1}{\int Pr(\boldsymbol{\theta}_2|M_2)Pr(\boldsymbol{D}|\boldsymbol{\theta}_2, M_2)\, d\boldsymbol{\theta}_2} \tag{4.3}$$

where $Pr(\boldsymbol{D}|M_k)$ is the marginal likelihood for model k.

Using the Bayes factor we can compare two models and will choose model 1 (M_1) if $B_{12} < 1$, and model 2 (M_2) if $B_{12} > 1$. There is an accepted scale of importance based on the magnitude of the Bayes factor, but as we must choose 1 model of the two we shall always use the favoured model, regardless of the level of importance.

Using the Bayes factor

We can make use of several simplifications which actually result in an easily calculated value. The Bayes factor can then be reduced to a simple factorial equation which is in fact much easier to calculate than the BIC and does not require calculating MLEs. This allows more models to be tested with the same computing power, or less power for any already being tested.

Given the integral for the marginal likelihood of the model and our two models based on the Dirichlet distribution, we can see that the prior distributions $Pr(\boldsymbol{\theta}_1|M_1)$ and $Pr(\boldsymbol{\theta}_2|M_2)$ will be equivalent and as such will cancel in the ratio/fraction B_{12}. This leaves us with the much simpler equation:

$$B_{12} = \frac{\int Pr(\boldsymbol{\theta}_1|M_1)Pr(\boldsymbol{D}|\boldsymbol{\theta}_1, M_1)\, d\boldsymbol{\theta}_1}{\int Pr(\boldsymbol{\theta}_2|M_2)Pr(\boldsymbol{D}|\boldsymbol{\theta}_2, M_2)\, d\boldsymbol{\theta}_2} = \frac{\int Pr(\boldsymbol{D}|\boldsymbol{\theta}_1, M_1)\, d\boldsymbol{\theta}_1}{\int Pr(\boldsymbol{D}|\boldsymbol{\theta}_2, M_2)\, d\boldsymbol{\theta}_2} \tag{4.4}$$

with $\int Pr(\boldsymbol{D}|\boldsymbol{\theta}_k, M_k)\, d\boldsymbol{\theta}_k = \int \pi(\boldsymbol{\theta}_k|M_k, \boldsymbol{D})\pi(\boldsymbol{\theta}_k)\, d\boldsymbol{\theta}_k$, i.e., the posterior probability distribution.

This posterior probability is a Dirichlet distribution such that:

$$f(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{n}) \propto \prod_{i=1}^{l} x_i^{\alpha_i + n_i - 1} \times \text{const.} \tag{4.5}$$

with the constraints $\sum x_i = 1$, and the actual integral being

$$\int \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{l} x_i^{\alpha_i + n_i - 1}\, d\boldsymbol{\theta}_k \tag{4.6}$$

$$= \frac{1}{B(\boldsymbol{\alpha})} \int \prod_{i=1}^{l} x_i^{\alpha_i + n_i - 1}\, d\boldsymbol{\theta}_k \tag{4.7}$$

$$= \frac{1}{B(\boldsymbol{\alpha})} B(\boldsymbol{\alpha} + \boldsymbol{n}) = \frac{\Gamma(\sum_{i=1}^{l} \alpha_i)}{\prod_{i=1}^{l} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{l} \Gamma(\alpha_i + n_i)}{\Gamma(\sum_{i=1}^{l} \alpha_i + n_i)} \tag{4.8}$$

We can show that the integral above evaluates to the Beta function $B(\boldsymbol{\alpha} + \boldsymbol{n})$ by using the fact that a pdf will integrate to 1:

$$\int \frac{1}{B(\boldsymbol{\alpha} + \boldsymbol{n})} \prod_{i=1}^{l} x_i^{\alpha_i + n_i - 1}\, d\boldsymbol{\theta}_k = 1 \tag{4.9}$$

$$\int \prod_{i=1}^{l} x_i^{\alpha_i + n_i - 1}\, d\boldsymbol{\theta}_k = B(\boldsymbol{\alpha} + \boldsymbol{n}) \tag{4.10}$$

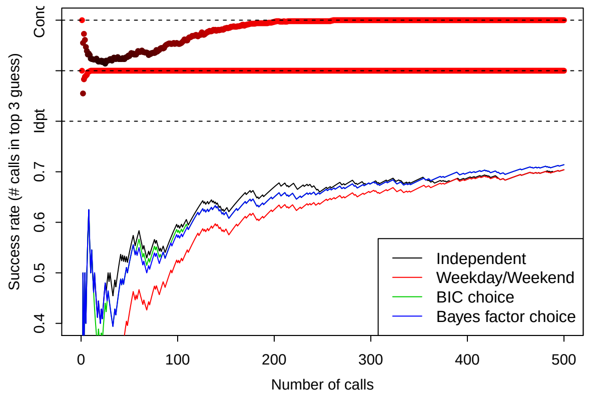


Figure 4.4.: 100 averaged simulations with the Bayes factor used to select model. Average BIC value and decision is indicated above the success rates between Idpt–Cond with BIC above Bayes factor. C.f. Fig 4.3

This gives us the result that

$$\begin{aligned} B_{12} = \frac{\int Pr(\boldsymbol{D}|\boldsymbol{\theta}_1, M_1)\, d\boldsymbol{\theta}_1}{\int Pr(\boldsymbol{D}|\boldsymbol{\theta}_2, M_2)\, d\boldsymbol{\theta}_2} &= \frac{B(\boldsymbol{\alpha} + \boldsymbol{n_{(1)}})}{B(\boldsymbol{\alpha})} \times \frac{B(\boldsymbol{\alpha})}{B(\boldsymbol{\alpha} + \boldsymbol{n_{(2)}})} \\ &= \frac{B(\boldsymbol{\alpha} + \boldsymbol{n_{(1)}})}{B(\boldsymbol{\alpha} + \boldsymbol{n_{(2)}})} \end{aligned} \tag{4.11}$$

where $\boldsymbol{n_{(1)}}$ is the vector of n_i for model M_1 and $\boldsymbol{n_{(2)}}$ is the vector of n_i for model M_2.

As we know that $\boldsymbol{\alpha} + \boldsymbol{n}_{(i)}$ will always be an integer we can use a simplify the Beta function as

$$\mathrm{B}(\boldsymbol{x}) = \frac{\prod_{i=1}^{l}\{(x_i - 1)!\}}{(\sum_{i=1}^{l}\{x_i\} - 1)!}. \tag{4.12}$$

4.5. Results

When calculating the Bayes factors for our models we will quickly reach a numerical limit in the product of factorials, e.g., if 20 calls have been made to a single recipient the Beta function will include at least one value of $(20-1)!$ which is already greater than 10^{17}, most systems cannot handle integers greater than around $2 \times 10^{1}9$, sometimes only 4×10^{9}. If we takes logs of the Bayes factors we will have a 1-to-1 mapping $(\mathbb{R}^{+} \to \mathbb{R})$ and as the log function is strictly increasing we can simply choose the model with the smallest $\log(B(\boldsymbol{\alpha} + \boldsymbol{n}_{(k)}))$. Taking logs also leads to a few more simplifications by allowing us to use the Binomial coefficient in place of the factorials, we have used the simplified equation below in the simulations:

```
sum(log(q)) + sum(log(v+1)) - (length(v)-1)*log(2)
                                 - log(sum(v)) - log(sum(v)+1)
```

with suitable checks for invalid values.

Figure 4.4 uses the same simulations as Figure 4.3. We have used the Bayes factors to choose the best model for predictions and added these results to the graph. We see that the Bayes factors suggest the weekend model initially, with the same sharp switch to the independent model, but returning immediately back to the conditional model again. The result of this is a prediction that is only just below the BIC choice and which tends to the same decisions as the BIC. Again, the choices made vary greatly between individuals as the importance of the conditional model also varies.

We can confirm that the Bayes factor can be computed far more efficiently than the BIC and from several simulations it appears to perform reasonably well.

5. Results and Conclusions

We have been fortunate to have access to a large quantity of real data. To access the content of the data we have investigated various approaches. Ultimately we found the most successful approach was to prepare SQL statements, trial them on toy data sets, and then run them on the SQL server where another table would be created with the results. This technique provides a repeatable solution which doesn't tie up local resources and allows for later reference and adjustments to the code.

Although large in quantity, the data-set is not necessarily rich in information with only a very limited range of information available for each call. Some careful processing of the complete data set allowed extraction of key features and some basic visualisations of the data. Various sample approaches were used, based on the summary statistics obtained, to provide a representative sub-set of the data. Features discovered at later points in the project were compared to the original data whenever possible.

Visualisations of the data have proved to be very insightful, care has been required when looking at overviews of the data as small features can be easily missed on plots with a large range. Such features may not be included within the usual areas of interest when examining data with a larger scale. An example would be the small peak at 3600 seconds in Figure 2.2c which was missed in the initial investigations. Visualisations were key to discovering the true ranges of some of the data fields where standard statistics alone such as means, modes, and inter-quartile ranges, would not easily describe the full extremes of some user's values. See Wilkinson [2005] for detailed investigations into visualising data.

A major part of the project was the development of analytical models for the call distributions. The independence model provided a useful base-line against which to compare our other models. The data showed that there was a significant difference between the likely next call on weekdays and at weekends and so a conditional model was required to capture this. We also investigated the importance of this condition when predicting the next call from a list of variable size. The size of this list, represented in Figure 3.3, can be used to determine the importance of further factors in conditional models.

The work explored the probability of correct prediction if one number was chosen and then two, three, or four numbers. It could be a valuable contribution to handset efficiency or the mobile network management if information was available saying "the next call will be to one of these two numbers"; resources could then be set aside for one call and network routes could be prepared to the possible recipients. Looking at a sample of individuals, implementing a list size of 3 to 5 on a device seems to give a high probability of predicting a correct call, while also not overcrowding the display.

To explore the prediction algorithms and to investigate the use of Bayesian information criterion for switching between two models of call generation, a large number of simulations were performed from a simulated data set. We have aggregated a huge number of calls, and focused on a small number of individuals when investigating our models, choosing some individuals as a basis for our simulations. Not all individuals' patterns of usage are suited well to some aspects of the predictive models, particularly when considering conditional factors with low usage users. In these cases the model switching algorithms can be used to make predictions based on the simplest models. In general the Bayes factors can be used effectively, but we may wish to continue using the BIC with new individuals and factors as it appeared to be less aggressive in switching.

Figure 4.3 and 4.4 show that model switching can perform well and with the greatly simplified equations for the Bayes factors we can compute predictions with very low memory and processing requirements, needing only a summary of the calls to each recipient under our conditions. We believe that this method is an improvement on the predictive model proposed by Barzaiq and Loke [2011] as we do not need to store the entire call history to generate the predictive distribution.

6. Further Work

6.1. Clustering

To improve our conditional model we can choose alternative factors. We may also desire to condition on more than two levels, e.g., 7 days of the week. As visible in Figure 2.3b we can see patterns in the hourly call volumes. The data in the graph has already been aggregated into hourly values, but this may still provide too many levels to produce a conditional model. We know that even two levels causes an initial reduction in the quality of predictions while enough data is gathered.

Predicting calls from continuous time also raises more difficult questions, we propose that cell towers could be used as a proxy for time as people will tend to be at work during 9–5 for example.

Considering up to 2,500 cell towers for each recipient on a data set with far fewer calls than conditions would not give us any useful predictions. A reasonable assumption is that some cell towers may have similar call distributions, possibly because they are physically close to each other. The same assumptions can be made about time, as already suggested, 9–5 on a weekday could have a single distinct distribution.

Assuming any fixed discretisation could have a detrimental effect on our predictive models and certainly could not be used for all users. Cluster analysis would allow us to dynamically determine which hours of the day, or cell towers to collect into a single distribution to reduce the number of conditions required for prediction.

Preliminary investigations have shown that there are indeed some patterns visible between different groups of cell towers.

6.2. Alternative directions

Determining important factors We can already use the technique of determining the portion of the predictive list that is constant between factors to measure importance. Isaacman et al. and Farrahi and Gatica-perez [2009] among many others are already using historical call data to categorise personality traits and important locations in people's lives. See also Burns et al. [2011].

Creating a system with weighted models We could consider producing a self-updating system that weights conditional models on their historical success rate. This would allow us to produce multiple models, each with a small number or parameters and use them together as a mixture distribution.

Predictability of people We may wish to determine how predictable different categories of people are and we would use cluster analysis to form groups of users suitable for different predictive factors.

Other uses of the data There are many more questions that can be investigated with this data by inferring certain attributes and combining with external data sources. Some such questions could be cover the effect of weather of social communications, or the difference in important predictive factors between individuals of different regional areas.

Public visualisations Activity levels (of phone usage or otherwise) can be an important indicator of important events in an area, providing easily accessible graphics and animations of this data in real-time could be extremely valuable or useful to end-users. A programming language such as Processing can be used to produce interactive environments.

A. Resources

A.1. Summary of tools and resources used

A.1.1. Tools

R is used to analyse the initial and resultant data, as well as perform the simulations and produce most of the graphs visualisations. `http://www.r-project.org/`

RevoScaleR is a commercial package for R that handles the high-performance XDF file format. `http://www.revolutionanalytics.com/`

MySQL is a popular open-source database, we used MySQL and the Structured Query Language to access the data while on the server. `http://www.mysql.com/`

MrSidDecode is part of the Unified SDK available from LizardTech, this software allows decoding of the MrSID files used to store the high resolution satellite imagery. `http://www.lizardtech.com/`

R packages used include `raster`, `sp`, and `maptools` as available on the CRAN repository.

GIMP and Inkscape are open source image editors for raster and vector graphics respectively. `http://www.gimp.org/`, `http://inkscape.org/`

Processing is an open source programming language used for creating images, animations and interactions. `http://processing.org/`

A.1.2. Resources

DivaGIS provide links to several websites as well as providing downloads for a huge amount of GIS data. We have used data for population density, transport routes, and administrative boundaries from these sources. `http://www.diva-gis.org/gdata`

NASA provide high resolution satellite photography of the earth circa 1990/2000. `https://zulu.ssc.nasa.gov/mrsid/`

References and Bibliography

David H Bailey and Jonathan M Borwein. Experimental Mathematics : Recent Developments and Future Outlook. pages 1–18.

The World Bank. The World Bank. URL `http://www.worldbank.org/`.

Osama O. Barzaiq and Seng W. Loke. Adapting the mobile phone for task efficiency: the case of predicting outgoing calls using frequency and regularity of historical calls. *Personal and Ubiquitous Computing*, 15(8):857–870, June 2011. ISSN 1617-4909.

Joshua E Blumenstock, Dan Gillick, and Nathan Eagle. Who ' s Calling ? Demographics of Mobile Phone Use in Rwanda. *Artificial Intelligence*.

Michelle Nicole Burns, Mark Begale, Jennifer Duffecy, Darren Gergle, Chris J Karr, Emily Giangrande, and David C Mohr. Harnessing Context Sensing to Develop a Mobile Intervention for Depression. *Journal of Medical Internet Research*, 2011. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3222181/`.

Joseph E Cavanaugh. 171 : 290 Model Selection Lecture VI : The Bayesian Information Criterion. 2009.

DIVA-GIS. Free Spatial Data. URL `http://www.diva-gis.org/gdata`.

Nathan Eagle, Alex Sandy Pentland, and David Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences of the United States of America*, 106(36):15274–8, September 2009. ISSN 1091-6490.

Katayoun Farrahi and Daniel Gatica-perez. Learning and Predicting Multimodal Daily Life Patterns from Cell Phones. *Time*, 2009.

Leon Gu. Dirichlet Distribution , Dirichlet Process and Dirichlet Process Mixture Binomial and Multinomial. Technical report.

Brent Hecht, Lichan Hong, and Bongwon Suh. Tweets from justin bieber's heart: the dynamics of the location field in user profiles. *Electrical Engineering*, (Figure 1), 2011.

Yanqing Hu, Jiang Zhang, Di Huan, and Zengru Di. epl draft Toward a General Understanding of the Scaling Laws in Human and Animal Mobility arXiv : 1008 . 4394v3 [physics . bio-ph] 31 Oct 2011. *Complexity*, pages 1–8.

Sibren Isaacman, Richard Becker, Margaret Martonosi, James Rowland, and Alexander Varshavsky. Identifying Important Places in People ' s Lives from Cellular Network Data 1 Introduction. *New York*, pages 1–18.

Kass and Raftery. Bayes Factors, 1995.

David Mohr and Marla Paul. A Therapist in Your Pocket. URL http://www.northwestern.edu/newscenter/stories/2012/02/therapist-phone-mohr.html.

NASA. Applied Science and Technology Project Office. URL https://zulu.ssc.nasa.gov/mrsid/.

Wei Pan, Nadav Aharony, and Alex Sandy Pentland. Composite Social Network for Predicting Mobile Apps Installation. *Artificial Intelligence*, 2011.

Andrew Pavlo, Erik Paulson, Alexander Rasin, Daniel J Abadi, Samuel Madden, M I T Csail, Michael Stonebraker, and David J Dewitt. A Comparison of Approaches to Large-Scale Data Analysis. 2009.

Christy Pettey. Worldwide Mobile Connections Will Reach 5.6 Billion, 2011. URL http://www.gartner.com/it/page.jsp?id=1759714.

Santi Phithakkitnukoon and Teerayut Horanont. Activity-aware map: Identifying human daily activity pattern using mobile phone data. In *Inter. Conf. on Pattern Recognition (ICPR 2010), Workshop on Human Behavior Understanding (HBU)*, pages 14–25, 2010.

David Smith. Accessing databases from R, 2011. URL http://blog.revolutionanalytics.com/2011/05/how-to-access-databases-from-r.html.

Chaoming Song, Tal Koren, Pu Wang, and Albert-László Barabási. Modelling the scaling properties of human mobility. *Nature Physics*, 6(10):818–823, September 2010. ISSN 1745-2473.

V Soto, V Frias-Martinez, and J Virseda. Prediction of socioeconomic levels using cell phone records. *User Modeling, Adaption*, (1), 2011.

Weather Underground. Weather History & Data Archive. URL http://www.wunderground.com/history/.

Leland Wilkinson. *The Grammar of Graphics (Statistics and Computing)*. 2005.

Huiqi Zhang and Ram Dantu. Predicting Social Ties in Mobile Phone Networks. *Science*, pages 25–30, 2010.