VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

Record linkage on Māori land data to find and connect the missing

Valerie Chan

Supervisors: Marcus Frean, Sydney Shep

31st July 2020

Submitted in partial fulfilment of the requirements for
Master of Computer Science.

Abstract

We consider probabilistic modelling for accomplishing record linkage across two large scale publicly available data sources: New Zealand Births Deaths and Marriages (BDM), and Māori Land Online (MLO). We undertake this project in the context of te ao Māori, integrating mātauranga Māori principles into the work. We present several methods for record linkage and several novel ways to reject false linkages.

Acknowledgement

An enormous and heartfelt thank you to Dr. Marcus Frean, who has been on this journey since the beginning. I have learnt so much from you.

I would like to thank Dr. Sydney Shep and the rest of the Wai-te-ata Press whānau, for the privilege of working alongside you on this project.

Thank you to the Parininihi ki Waitotara for supporting this research and welcoming me into the project. Thank you for introducing me to Tikanga Māori, inviting me into your marae and for sharing your stories. It has been an honour and a privilege.

I would like to acknowledge the tipuna who have come before, and who are present in the stories and the data that I have studied.

I gratefully acknowledge the financial assistance provided by the National Science Challenge. This assistance was a part of a Science for Technology and Innovation grant for the Spearhead 4, Te Tātari Raraunga/Māori Data Science.

I would also like to thank my family, who are my support system and provide both a supportive ear and a helpful eye.

Taonga

When working on this project, we have been careful to respect the cultural significance of the data and the stories that have been shared. These are Taonga, meaning that they are a treasure or something prized. The stories and the data are treasured because they contain the names of kaumātua (elders). A reader who continues to read this thesis agrees to not use the data shown without first obtaining permission from Parininihi ki Waitotara.

Contents

Figures

Rere ki uta
Rere ki tai
Tau mai te manu
Pitakataka ki to pae e -
Fly inland
Fly coastward
The bird settles
And flits about its perch

Waiata [Author Unknown]
Translation by Mitchell Ritai. Paraninihi ki Waitotara

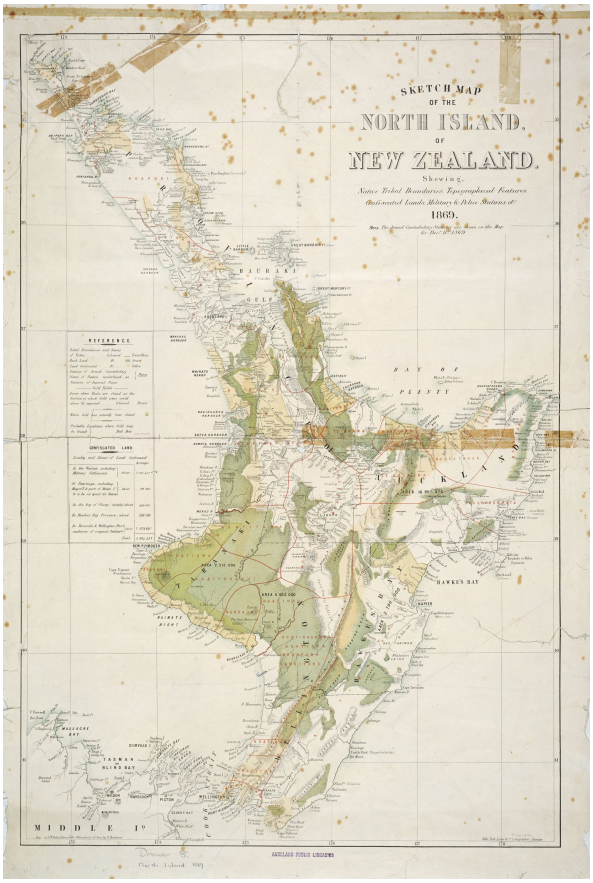


Figure 1: Te Ika a Maui | North Island, New Zealand ([1]).

Ko Ruapehu te māunga
Nō Te Whanganui a Tara ahau
Haere mai toku tupuna i Haina
Ko Valerie tōku ingoa
Tēnā koutou, tēnā koutou, tēnā koutou katoa.

-

The mountain that I affiliate to is Mt Ruapehu
I am from Wellington
My family is from China
My name is Valerie
Greetings, greetings, greetings (to all of you).

Chapter 1

Introduction and Motivations

The impact of nineteenth-century Māori land confiscations continues to be felt in modern day New Zealand. Despite partial restitution and treaty settlements, the task of reconnecting with the descendants of the Māori who originally lived on the land is almost impossible. The task of identifying, contacting and engaging with these missing descendants constitutes an enormous challenge for Māori incorporations, iwi and hapū. There is a multitude of issues with this process, starting with the lack of accurate data, lack of existing tools and difficulty of land succession. The follow on effects are huge and have meant that many opportunities for social and economic improvement have been missed.

This thesis sits within a wider multidisciplinary research project *Kimihia te Matangaro - Finding the Missing* created in response to a Science For Technological Innovation National Science Challenge grant (SFTI)[2]. The purpose of this challenge is to tackle New Zealand's technological challenges with a "focus on building enduring partnerships between researchers, business, and Māori organisations". This project is a collaborative project between *Parininihi ki Waitotara* (*PKW*) [3], *Victoria University of Wellington* and *Auckland University*. Māori-owned business *PKW* was set up in 1976 to manage the assets on behalf of the 5396 original owners[3].*PKW* has built a thriving operation in Taranaki investing in many diverse operations from forestry and farming to commercial property. These investments return dividends and should be shared with the (now) 10,000 plus shareholders of the incorporation. However, over time the connection with many of these owners has been lost. This means that there is a large volume of unclaimed dividends that needs to be distributed to the missing *PKW* whānau. The National Science Challenge project was to work with *PKW* to provide technological solutions to help with the identification process.

From the very beginning, this project it has been grounded in understanding the problem in the context of te ao Māori (i.e. with a Māori worldview). The interrelationship between whānau, whenua, and te reo frames all of the engagements with the project members and the project data. Research in this thesis has been undertaken within the social and cultural realities of mātauranga Māori and traditional western science frameworks.

1.1 Whānau and Whenua

The land and the People

Before the European land ownership models we use today were introduced, Māori land was held collectively by the iwi or hapū, and rights to occupy land was determined by group. A person's whakapapa (*genealogy*) ties them to the original occupiers of the land so they were provided with the rights that go with the land. When the British began colonising

New Zealand the treatment of the Māori indigenous people and their land was not always (or often) fair. Despite The Treaty of Waitangi being signed between the Crown and a number of Māori chiefs across the country beginning on 6 February 1840, the mid 1800s in New Zealand was characterised by bloody skirmishes between Imperial Britain and Māori. The British wanted to acquire more land and the Māori did not want to give up what was theirs. This is commonly referred to as the New Zealand land wars [4]. By the end of this time, millions of hectares of Māori land had been confiscated by the Crown. Although the fighting had largely come to an end, the period of the late 1800s to the early 1900s marked even more significant land confiscation and alienation for Māori. Instead of through fighting, this was achieved through forms of legislative and bureaucratic means [17].

The Native Land Court Act was established in 1862 in order to allow European settlers to purchase Māori land. This caused considerable land loss and alienation of Māori from their land because they often had to use sections of their land as down payments for food and travel costs to get to court hearings across the country[46]. An example of land loss can be seen in Figure 1.1 which shows how significant the area of confiscated land was. Since the certificate of title could not be issued to more than 10 people, there were many land disputes where land had belonged to larger groups that persist still to this day, and absentee ownership is common [17].

Each Māori incorporation maintains its own separate share register which is a list of Māori who are shareholders of various land blocks in the domain of the incorporation. The lists contain names of Māori collected by the incorporation over the last 50 years. These lists were not always compiled accurately, and (more seriously) in some cases only contain names and no other identifying information about the individual. Some of the lists are handwritten with inconsistent formats and the listed names may not even be an individual's official name. There are often duplicates and omissions. Many of the lists have not been updated to reflect transfer of ownership from original shareholders to their descendants. A cumulative effect of all these imperfections is that there is now a significant proportion of eligible people who are not aware of their shareholder status or their connection to the land who cannot be contacted. Māori Land organisations (like *PKW*) allocate the dividends from the land that they manage to their shareholders. The task of allocating this money to the right people relies on a combination of data sets, as well as local knowledge to find their shareholders.

The main data set is Māori Land Online (MLO) which holds the results of Māori Land court hearings. Māori Land Online [36] and Land Information New Zealand (LINZ) [37] have geographic data which shows which shareholders are eligible for which blocks and where the blocks are located. The problem with the Māori Land Online data set is that the data is typically out of date. This is for several reasons, the first is the complex inheritance process and the second is the typically low financial pay off. Inheriting shares has an application process and requires going to the Māori Land Court. This must be done for each block of land that the descendant is inheriting. There is often very little monetary incentive for someone to complete the application, as the original shares have been divided into smaller values as they are passed down through the generations. For example a land block may return as little as five dollars per year.

Part of the identification process involves finding 'missing' individuals from a land block in another official database so that contact might be made. However many of the names that are listed in MLO/Māori Land Court records are not easily found in traditionally used genealogy data sources such as the New Zealand Births Deaths and Marriages records (BDM). In principle, BDM would make a good data source for identifying individuals, as it is well indexed and can be assumed to be complete. However it is an expensive data set (payment being required per query) which may limit its use in practice for high volumes of data. The

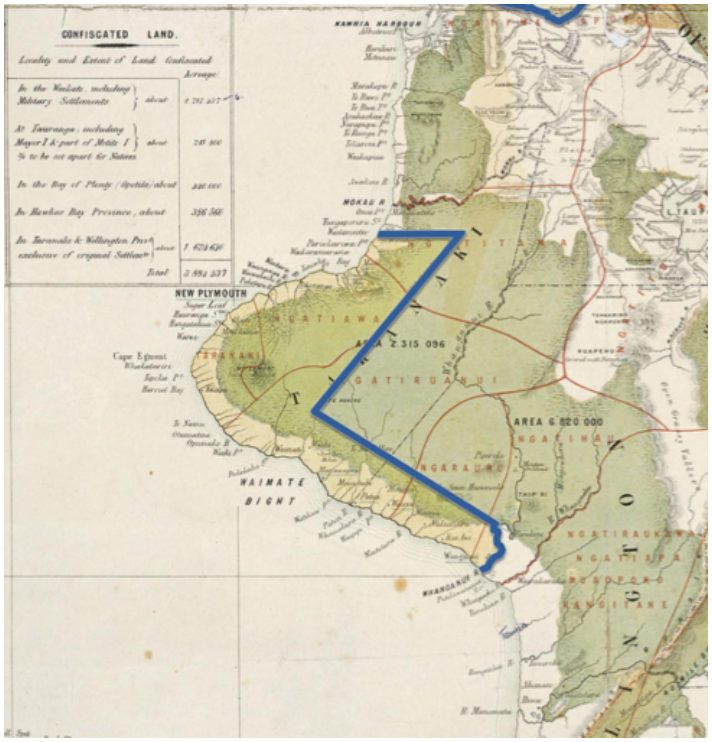


Figure 1.1: **Taranaki confiscation line.** This map of the Taranaki is from the late 1860s. The boundaries (in blue), which were added later, show the main area of land confiscated by the Crown after the New Zealand wars of 1860. Smaller parcels of land outside this main area were also confiscated.

([5]).

freely available subset of this data source, Births Deaths and Marriages historic (BDM-H) does not have any data from the last 50-100 years in order to protect the privacy of living people. It does not seem therefore, to be a useful resource. However the individuals from the original lists and other historic shareholders are likely to have birth records inside of the time period that is not available on the historic site [33]. One of the aims of this project is to understand how useful this data is by itself. The wider project may be able to gain access to the full data set given that this project is of cultural and social importance. Part of this project will be to assess the potential value of having this data.

In addition several of the iwi groups have their own whakapapa (*genealogy information*). In the past "whakapapa was the way Māori proved they had links to a piece of land when they went to the Native Land Court to try to gain title to the land." [6]. PKW have access to (some of) these through their local community, however the project is not allowed to view these records. While Māori whakapapa is often known and performed as a waiata (*song*), the written form is considered tapu (*prohibited*)[6]. These are concerns data management, ownership and digitalisation.

The cumulative effect of the idiosyncrasies of the data sources available means that finding descendants of original shareholders may be very difficult. The task of manually searching for hundreds or thousands of names from the original lists is potentially fruitless. Given the significant length of time and potential impossibilities of manually searching for these people this problem presents well as a candidate for machine learning or inference. While there are many potential data sources, the majority of these are private (whakapapa) or prohibitively expensive to access (BDM), and none of them contains "ground truth" - the actual mapping between names and identities. Because of the low data quality a shareholder's name alone is not a unique identifier: a single shareholder may have been entered in the data set with more than one identifier or name (think maiden names, married names, spelling mistakes), and conversely the same name may be in use by multiple identities. Toy data sets were used in initial stages of the project in order to focus on the inference/machine learning problems as opposed to the data acquisition problem. As the larger project investigates a wider net of real data sources, these were shared between project members. It is also important to ensure solutions are flexible and applicable to different groups, as other incorporations within New Zealand are facing similar problems.

1.2 Goals

The task of identifying and contacting likely present-day shareholders, based on lists of names that contain ambiguities and may be several decades old, admits a variety of approaches. The aim of this project is to explore options for an approach that leverages hitherto unused large-scale data sources, such as BDM, Māori Land Court records, MLO, LINZ [37] and whakapapa, (where available) in service of the overall task.

1.2.1 Objectives

The objective of this Masters was to investigate solutions to identify the missing shareholders. This will be done by identifying the key inference step or steps, and then developing culturally appropriate methods. This will allow us to apply probabilistic inference where appropriate to find missing shareholders.

Specifically this project aimed to:

- Analyse the key inference processes that are required when using a range of data sources to identify potential shareholders.

- Analyse the key inference processes that are required when generating new information that is likely to be of value to someone searching for present-day shareholders.
- Identify key sub problems that could be solved using machine learning or inference that are in line with the project's goal.
- Develop a sandbox for testing possible solutions for a variety of inference processes and data sets.
- Generate or source toy data sets to be used initially. Assess how well the inference model performs on the data sets.
- Document use of the sandbox.
- Evaluate potential techniques for establishing matches.
- Evaluate potential data and feature sets (MLO BDM, LINZ, whakapapa).

A stretch goal for the project was to explore how the developed process handles real data. This would be dependent on access to data and sufficient success on toy data.

1.3 An outline of the thesis

First the use and inclusion of the Māori world view in scientific research projects is discussed. This provides a cultural understanding of how important it is to be respectful, especially when operating as part of a kaupapa led research group. We also look at existing research in the field of Record Linkage. With the ultimate goal of being able to identify and find missing PKW shareholders we are focusing on linking groups of individuals from MLO and BDM. These groups can be thought of as sibling groups. We are trying to find a family of shareholders from MLO in BDM as we believe that this will provide a legal name. We introduce two novel methods for doing this: group-to-group and alignment specific. Group-to-group linkage looks at each group of shareholders as a whole, we look at how good the BDM group is at generating the MLO group. Whereas the alignment specific approach matches each individual with another individual and using dynamic programming creates an alignment between the two groups. We introduce various methods to reject linkages that we do not believe contain a real link. We have applied these linkage methods to test data, MLO and Cenotaph records.

Chapter 2

Background

2.1 Respecting the Māori worldview when undertaking research involving Māori

mātauranga Māori - **Māori knowledge** - the body of knowledge originating from Māori ancestors, including the Māori world view and perspectives, Māori creativity and cultural practices. [8]

It is only right when researching in the Māori domain to understand the systemic and widespread effect that colonialism has had on our indigenous people. As part of this research we must understand the Māori world and the legitimacy Māori traditions and realities. We must take into account, and understand, the history of Māori experiences with non-Māori researchers.

Western science is frequently applied to Māori people and Māori data with little or only token efforts to understand the belief systems and values of the Māori community. There are some key Māori concepts that can act as a guide to researchers. Fiona Cram (2001) in her work on the validity and integrity of Māori research outlined some key Māori values that are important when entering into a research process in a respectful way [7]. This framework was developed specifically from a Māori perspective. It provides researchers with a set of values that inform research practices and ethical processes.

- *Whanaungatanga* - **relationship, kinship, sense of family connection** [9]. In a research project this means establishing meaningful, reciprocal and familial relationships in culturally appropriate ways. Being more engaged creates a deeper commitment to other people.

- *Aroha ki te tangata*-**To love / a respect for people.** Within a research context this value is about allowing people to define the research context and allowing them to define the terms of the interactions [13]. It is also about maintaining this respect when dealing with research data [11].

- *Manaaki ki te tangata* - **hospitality, kindness, generosity, support** - the process of showing respect, generosity and care for others [10]. In a research context this means involving the people who you are studying right throughout the process and sharing the results with them. The research should be for them, and they should receive closure when it is done [13].

- *Mahaki* - **Mahaki relates to humility.** Working to find ways to share and to be generous with knowledge. "Sharing knowledge is about empowering a process, but the community has to empower itself"[7]. Importantly Mahaki means sharing and empowering each other without being a show-off or being arrogant.

- *Mana* - **prestige, authority, control, power, influence, status, spiritual power, charisma - mana is a supernatural force in a person, place or object** [12]. In particular the Māori saying: *Kaua e takahia te mana o te tangata* which translates to *Do not trample on the mana or dignity of a person* is an explanation of how to handle oneself in research. This is about informing people and guarding against being patronising or impatient because people do not know what you know [7].

- *Titiro, whakarongo, korero* - **To look, and listen first and then maybe speaking** [13] . This value emphasizes to researchers the importance of understanding the context of a situation in order to develop understandings and find a place from which to speak [7].

- *Kia Tupato* - **Be cautious** [13]. Researchers need to understand the importance of being politically astute, culturally safe and reflective about their status as an insider/outsider. This is also a caution to others in the community to be frank with the researcher as the researcher will not know or be aware of everything [7].

Western science involving research on, with, and/or for people often involves the gathering of information. This may be done for its own sake but has historically been done with a view to informing resource allocation and facilitating control. "Research is therefore about power and power commands resources" (Te Awekotuku, 1991). "Māori research, by, with and for Māori, is about regaining control over Māori knowledge and Māori resources. However such research is not done in a vacuum – in the past non-Māori researchers have committed many transgressions against Māori. This has led to suspicion and a lack of trust of research within Māori communities" [11].

Māori have historically been denied sovereignty within colonial and western processes. An example of this can be seen in the 2018 New Zealand census which was taken online. Previously paper forms had been handed out and collected in person by thousands of census workers across the country. The online census has disadvantaged Māori participation and the response rate was as low as 80%. There were even lower response rates in the areas such as Northland and the East coast where responses are typically poor even in a normal year. The effect of poor response may have significant long term impact on Māori in New Zealand and create bias if the census results do not statistically reflect the New Zealand population. "If Māori descendants are missing in large numbers from Census 2018, this will reduce the size of the Māori electoral population and, potentially, the number of electorates. The census count of the Māori descent population is part of a statutory formula used to determine the boundaries and number of Māori electorates." [47]. The census is the primary way of getting data about Māori. The census is used by iwi and other government agencies for policy and planning purposes and also for Treaty settlements. There was low engagement with Māori in the change in census format. This is an important reminder to establish who will be affected by systematic changes such as this one.

Some studies have been done investigating appropriate methods for studying in a Māori context. In their work on ethical guidelines on health and disability research in relation to Indigenous People, Kennedy and Wehipeihana (2006) use Social Network Analysis in the context of indigenous and minority groups within New Zealand [16] . Social Network Analysis (SNA) is the study of the structure and composition of networks and is useful

for understanding the implications of relationships and patterns within social entities. The structure of the social ties, systematic data, graphic representation and mathematical or computational models are four key features of the SNA proposed by Kennedy and Wehipeihana [16]. SNA focuses on social context among all of the participants rather than on the rational choices made by each person.

Kennedy and Wehipeihana presented the following ideas for working respectfully with indigenous peoples to alleviate or eliminate barriers. Interestingly these are much in line with the work done by Cram [7] mentioned above.

- Self determination (the right to make decisions about all aspects of ones life);
- Clear benefits in participation;
- Acknowledgements and awareness of cultural values, customs, beliefs. That a world-view other than a western one exists.
- Cultural integrity- knowing that cultural knowledge must be protected from misuse and misappropriation and preserved for future generations.
- Capacity building, enabling indigenous people to participate actively in research with the aim for them to ultimately drive it themselves.

There are some examples of positive blending of western science and mātauranga Māori. In the 52nd edition of the New Zealand Journal of Marine and Freshwater Research, the editorial team Clapcott et al (supported by kaumātua *Rauru Kirikiri* who provided a cultural safety to the process), incorporated and adopted kaupapa Māori principles to the standard journal peer review process. This is very appropriate as the relationship to the marine and freshwater environment is highly significant to Māori. This special issue was collated with the Māori world view in mind and closely aligns with mātauranga Māori principles [14]. The journal selected papers using the guiding principle of *Tino Rangatiratanga* "sovereignty, autonomy, control, self-determination and independence and allowing Māori to control their own culture, aspirations and destiny". The hope of this special issue was to initiate further discussions on science using Māori priorities. The editors also reflect that there is still an ongoing need to build and nurture cross-cultural understandings between Māori and western science. This is particularly in regards to the rights, interests, and values of Māori across institutions, agencies and researchers from a wide range of disciplines.

Another area where mātauranga Māori was used in conjunction with western science was by Moller in his 2009 paper on New Zealand Seabirds [15]. The paper utilised many of the same principals of mātauranga Māori that were used by Clapcott et al [14]. Moller notes the preferred use of Māori words in place of the English/Latin as would be normal western scientific convention. Researchers were able to talk to kaitiaki and understand their views on the environmental management of titi (sooty shearwater) breeding colonies in their domain. The study of titi is culturally important as well as being important ecologically. This is because the harvesting of the titi is the last remaining widespread customary bird harvest which remains almost entirely within the control of Māori. The purpose of the research and relationship with Māori was to "ensure that the tītī remained plentiful enough for the mokopuna (grandchildren) to be able to continue mutton birding" [15].

2.2 Record linkage

Record Linkage is the problem of identifying and linking multiple records that relate to the same latent entity. These records could be within the same data set, or across multiple data

sets. Duplicate data entries often occur when large administrative programs create records for individuals or objects and then the system is not maintained, for example when a new system is implemented and records are duplicated or mis-entered during the migration. This type of problem is not uncommon in records about people. Examples of well known systems that have been used in record linkage studies in the past are: hospital records, medical records, census data, social security data and taxation data [32].

It appears that there has been more research in recent years into the record linkage problem. It is not unsurprising that interest in the record linkage has become more significant over time. Increased digitisation would create more applications.

The field of record linkage is relatively small - with only a few key researchers working on the problem. Felligi and Sunteb were the first to propose a mathematical model for "one to one entity mapping" in 1973[18]. They completed a case study on 1973 census data of record linkage. The case study attempted to match an individual's social security number to the same individual's census record. This was difficult as over 22000 people had not submitted their census form with the social security number filled out correctly. A manual search would have produced high quality results but would not have been feasible due to the size of the database. Felligi and Sunteb developed a mathematical model to determine whether or not two records refer to the same individual [18]. Their model generated a probability of a match and then classified the two records into one of three states: a link, a non link or a possible link. The two records were compared, and categorised to one of the following states; "name is the same", "name is the same and it is example", "Name disagrees", "Name missing on one", "Agreement of address". These states were used to generate a comparison vector as a vector function of both records, then the linkage was classified.

The work of Felligi and Sunteb was not seriously extended until 2013 with Sadinle and Feinbergs' work to link records across more than two databases [26]. Sadinle and Feinberg consider the problem of obtaining non-transitive linkages across these multiple datasets. A transitive link would mean that when an individual in dataset A was linked to an individual in set B, anyone else that was subsequently linked to B, would also be linked to A. Steorts is the most recent contributor to the field of record linkage, publishing papers in 2014, 2015 and 2019 [19] [20] [21]. In 2014, Steorts considered the idea of using a clustering of records to an "unobserved latent entity" with a hierarchical Bayesian model [19]. In this approach, records from different databases are clustered together around the "real" person who is actually unobserved. The work from 2014 was limited as it could only be applied to categorical data, which means that data features such as names and addresses were considered to be categorical rather than continuous which they tend to be.

Steorts adapted the work done in her previous paper in 2014, to use an Empirical Bayesian model instead of a hierarchical model. This Empirical Bayesian method uses a prior generated from the empirical distribution of the data values in that field. This is actually a simplification of the hierarchical model as it removes the need to specify a prior for the latent entity which is quite difficult. Because there was no labelled data it would have been very difficult to set a prior on the latent entity.

Steorts (2015) [20] in her paper *Entity Resolution with empirically motivated priors* introduced a novel (at the time) method for performing record linkage. This predominantly focused on the application of record linkage across multiple databases (more than two). Previous approaches to linkage problems used pairwise comparisons which is "computationally infeasible" on most databases of even moderate size. Steorts viewed linking records as an "unsupervised problem of determining the edges of a bipartite graph that links the observed records to unobserved latent entities". She used statistical inference techniques from the Markov Chain Monte Carlo family to determine these links.

Records are compared by using matching functions. These matching functions often take

variation, typographical and numerical errors into account as well as making more complex distance or time comparisons using look-up tables (Christen 2006) [22].

Steorts and Tancredi in their 2019 paper *A unified framework for de-duplication and population size estimate* [21] they build upon Steorts' earlier work. The paper does not take into account the terrible quality of most datasets that applications of deduplication usually contain. An example of this is that the paper assumes that unique identifiers are correct. In our application with Māori land data we know this is not true as we have many examples of individuals who have multiple identifiers. Another assumption that this paper makes is that there is no missing or patchy data. This is a simplification that cannot be made in the Māori Land Online dataset. An extension would need to be made for realistic applications of this method.

The use case for this method is joining two datasets that have the same parameters or dimensions - which is applicable to comparing two versions of the same dataset but not useful to complete record linkage across multiple different databases with different fields. The paper's novel element is that the population size is an unknown parameter, which it was not in the original work by Steorts in 2015 [20]. This paper also found a 'more adequate prior distribution for linkage structure'. They mention a paper by Chen (2018) that investigates applications of these methods.

Most recently, Stringham (2020) [27] apply many of the previous Bayesian methods when linking US Census data from 1900 with Union Army recruitment data. A refinement of the work of Sadinle (2017) [26] "relaxes the assumption of a fixed comparison data model and allows for record-specific disagreement parameters conditional on non-match status".

Another method for increasing the accuracy in string matching is to use the phonetic encoding of words. This allows records with spelling mistakes, typographical errors or alternative spellings to be linked regardless [42].

2.2.1 Blocking

Blocking is a method of reducing the number of candidate records that are involved in a comparison to a feasible number [40], by involves indexing or filtering the records [39]. Blocking is an important technique because of the massive computational cost of comparing all of the records against each other. When taking two databases A and B, and trying to find the links or duplicates, every record from A must be compared to every record in B. This means that the number of comparisons between records is the product of the number of records in each database. If there are duplicates in the databases then for example each record in A needs to compared to all of the other records in A:

$$searches = \frac{|A| \times (|A| - 1)}{2} \tag{2.1}$$

Brute force comparison creates a significant bottleneck in the performance of linking solutions [28]. This becomes unfeasible when databases are large, linking datasets with millions of records can take hours or days of compute. Record linkage problems become infeasible to solve through a brute force approach when databases are large. The extension of record linkage to extend across more than two databases was unfortunately computationally infeasible as it required the estimation of $2^{(N-1)}$ conditional probabilities in a database with N records [40].

The benefit of using blocking can be easily displayed. Suppose in our dataset A from earlier, where there are $|A|$ records and x blocks (each of which is the same size). The number of comparisons in theory [39] becomes:

$$searches = \frac{|A|^2}{x} \tag{2.2}$$

The number of comparisons in 2.2, is significantly fewer searches than 2.1, because we are able to reduce the size of the search space.

In blocking there is a single attribute or feature, or a combination of attributes or features that is called the blocking key. An example of a blocking key is to use the first four characters of each record's surname attribute. This is used to split the full data set into blocks. Narrowing the search space - candidate pairs are generated from only other records in the same block. It is important to consider block size when selecting a blocking key as block size can affect both the speed and the accuracy of the linkage problem. If the block sizes are too large then searching becomes inefficient as many more candidate pairs are generated, leading to an increased number of comparisons. If the number of candidate records is too small the accuracy of the linkage decreases as the true candidate pairs may be missed.[41]. However in order to achieve good a high linkage accuracy it is best to use the least error prone record attributes available. A way to increase the accuracy of the blocking key is to create composite blocking keys, these are keys made up from more than one attribute such as name and age.

2.2.2 Supervised Record Linkage

In supervised learning, labeled data is used to train a model. Data is said to be Labeled when it is tagged. In the case of record linkage each labeled record is tagged with the corresponding record that it is linked to. This allows the researcher to know if their model's predictions are correctly identifying linkages.

It is (relatively) easy to classify identical records and obvious non matches but it is very hard to differentiate those on the border where the two records share some attributes. Supervised machine learning tactics have been used more recently to do identify linkages however the downside of this is that it requires training data. Training data is not always present in real scenarios.

2.2.3 Unsupervised Learning in Record Linkage

In unsupervised learning, unlabeled data is used to train a model. Data is said to be unlabeled when it is not tagged. In the case of record linkage, unlabeled data means that the researcher can never *truly* know if their model's predictions are correctly identifying linkages.

In his paper on unsupervised Record linkage, Christen [22] presents a method of generating data and then carrying out supervised learning. This relies on two assumptions: First, that when the weight vectors generated in the comparison of two records have high similar or exactly the same values in all their vector elements the two records refer to the same entity, as it is very unlikely that two different entities have high similarities in all their attributes. Second, that when the weight vectors generated in the comparison of two records have mostly low similarity values the two records are different as it is highly unlikely that two records that refer to the same object would have low similarity values.

Christen investigated whether it is possible to use initial weights of highly likely matches and non matches as training examples [22]. Several methods for selecting which weight vectors/record matches and non matches to use were evaluated. A threshold was used to select appropriately likely weights (usually with a value of 100% or 0%). Another approach was to select based upon the nearest vectors using the Manhattan distance between two

vector weights. (The Manhattan distance is : *"The distance between two points measured along axes at right angles. In a plane with p1 at (x1, y1) and p2 at (x2, y2), it is* $|x1 - x2| + |y1 - y2|$*."*[23]. A problem that the paper did not answer but did address was that the data in the two data sets is linearly separable (which real data is likely not). They suggest that adding some more randomised data would help this and be more like real data. 10 fold cross validation was used. A support vector machine (SVM) classifier was used, due to its ability to handle high-dimensional data and be robust to noisy data. They also tried a K-means clustering algorithm. They also tried an 'optimal threshold algorithm' which has access to all of the other match options and assigns matches based on the highest probability overall. SVM worked better than the optimal threshold model.The paper mentions Febrl which is a free open source record linking software developed in 2004 (also by Christen[22]) using python [24].

2.2.4 Semi Supervised Learning in Record Linkage

Semi-supervised learning is used in classification problems where some but not all of the records have corresponding class labels. The number of labeled data points is less than the number of unlabeled data points. Obtaining class labels is often expensive and difficult. A solution to this is to use bootstrapping. In bootstrapping the model is fed by its own predictions using the labels that are present. Bootstrapping can produce results far superior to unsupervised methods [45].

Kingma et al (2014) looked at using probabilistic semi-supervised techniques to generate models trained on small labeled data sets. They then generalised these models to work on larger unlabelled data sets [43]. They designed a successful classification model made up of two parts. First they used a deep generative model on the labelled data. This generative model provided an embedding of the data. By creating an embedding (and representing that data as lower dimensional vectors), they could improve the ability of the network to learn from text data. The second part was a generative semi-supervised model that used the embedding from the first generative model. A limitation of the algorithm was that the models scaled linearly to the number of classes in the data set [43].

Later in 2017, Lee et al performed inference tasks on Korean genealogy data. The researchers attempted to infer political power structures based on the human network. They tried to classify individuals by which one of the two political forces they supported. Semi supervised learning was used to label the network. They took labeled individuals and inferred similar political views onto their blood relatives. A similarity matrix was constructed to show edges (relationships) between people. The weights of similarity between nodes are calculated by Gaussian functions. They also suggest additional measure that any two connected nodes should not have a "high similarity difference" [44].

In 2018 Malmi et al in their paper *Computationally Inferred Genealogical Networks Uncover Long-Term Trends in Assortative Mating* matched genealogy data to church collected Births Deaths and Marrige records from mid 17th to the late 19th century Finland. They were doing genealogical network inference problem (population reconstruction) by linking the birth records of an individual to the birth records of the parents, creating a family tree with up to millions of individuals. They used the inferred network to look at assortative mating (marrying others within the same socioeconomic group).

Similarly to Lee et al [44] they had access to a genealogical network consisting of over 100,000 individuals. This was constructed by an single genealogist over a long period of time. They used this network as ground truth. An individual was considered matched between the genealogy and the birth records if they found exactly one birth record with the same normalized first and last names and the same birth date. They used this information

as training and testing data for their algorithm. Naive Bayes, Random forests and two variations of Binary classification were used to reconstruct the network. The binary Classification methods outperformed the other methods. Accuracy is in the mid 60s which is not ideal.

Since the data that was used as 'ground truth' in the work by Malmi et al [25] and Lee et al [44] is based upon human generated data there may be a mistakes or inaccuracies. Malmi et al accept the fact that this human error will then affect the performance of the algorithms and the conclusions drawn from any results. They also mention that generalisability of such a problem would be useful as many ethnic and historical groups have this type of problem.

2.3 What is missing?

The work done by Feligi and Sunteb, Steorts and Tancredi, and Stringham share two common weaknesses. The first is that they all focus on record linkage on high quality datasets. There is a reliance on having either good labelling of the data or high quality features for each record. This is a drawback to their research, as this makes it not applicable to many scenarios. Record linkage on low quality data has not been covered before.

The second weakness that previous researchers have in common is that they are trying to link individuals. There is no consideration of using natural groupings within the data to enrich their models. In a linkage scenario where there are many individuals with the same name it is hard to know which records can be linked.

An example of using natural groupings can be seen in the work done by Lee et al where they use the natural grouping created by blood relations to infer political opinions [44]. Had they not done this their task would have been almost impossible. The company that a person keeps can be used to infer linkages as there is evidence in the records around the individual.

Chapter 3

Finding the missing

We want to find people, but names are not a very good identifier. We instead aim to link together groups in MLO and BDM as we believe that natural groupings and connections between families might be better at identifying real individuals.

Names in isolation are only of limited utility. We do not know the real person behind the name or how to contact them. Names do not by themselves convey enough reliable meaning to be able to identify a 'real' individual. Māori names and English names were used interchangeably. Shep et al in their paper *Indigenous frameworks for data-intensive humanities: recalibrating the past through knowledge engineering and generative modelling* [46] provide an example of a fictitious individual *Erana*, "who also goes by *Ellen*, but appears in the birth records as *Sarah*. *Sarah* is not used by the family, when transliterated from the te reo Māori corpus is Hera, again unused. To complicate matters, Sarah Ellen appears twice in the official birth records with different registration numbers. Similarly, her brother `Himi` is also known as `Jimmy`, but according to the Crown, is legally `James`. While the linguistic distance from `Himi` to `James` can be quantified, `Himi` is aurally closer to `Jimmy`, whereas the more common `Hēmi` is closer to `James`, thus reflecting the mutability of oral and written exchanges between te reo and English" [46]. Currently *PKW* uses a combination of MLO data and their own records to identify shareholders who have money to collect. This process is expensive and often fruitless as necessitates working out who a person really is (finding their real identity) and then trying to contact them. Inferring such identities at scale across the whole of New Zealand is the challenge. This is made even more difficult through immigration to other countries - there are individuals born overseas who have no records in New Zealand who could be eligible to receive dividends.

Let us return to the idea of natural groupings that were introduced in the previous chapter. We know a lot about a person from the company that they keep. In a Māori context the concept of whānau, hapū and iwi (*family, community and tribe*) provide a basis for natural groupings within our data. In general we expect individuals to be on the same blocks as other individuals within their whānau and hapū. Since iwi groups are a lot larger it is perhaps less likely that we could use iwi as a method of filtering. We will be focusing specifically on an individuals close whānau with the natural grouping of interest being them and their siblings. Knowing who a person's siblings are is important. Siblings (in most cases) all inherit from the parents. This means that in a group of siblings, using the most basic rule of equal inheritance, in a family with N children, each child inherits $1/N$ of the land interests.

MLO provides us with more than just a name, we also have minute book references, share values, gender, alternative names and whether the owner has a trustee or not. This can be seen in Figure 3.1 as the different rows for each owner. Both the Minute Book reference and the share value provide us with a natural grouping. The minute book reference is a reference to the Māori Land Court hearing where the inheritance was processed. We are

able to see who else inherited ownership at the same time, by looking for other individuals with the same minute book reference on the block. This helps to identify who are likely to be siblings of an individual.

We can see an example of this in Figure 3.1 where the red dotted line outlines four individuals with the same share value from the same minute book court event.

We have access to the corresponding Minute Book reference (414 Aotea MB 282-283 dated 17 March 2020) and a copy has been included in the Appendix **??**. The shares came to `Andrew Clayton Robinson`, `Daniel Shayne Robinson`, `Joval Margaret Tamou` and `Taina Shaun Tamou` from `Robert Tamou`. The order is a determination of a life interest. The order shows that the shares came to `Robert Tamou` from his wife `May Bishop` (also known as May Tamou). Because it is a determination of life interest it indicates that `Andrew, Daniel, Joval` and `Taina` are children of `May Bishop`. PKW have historic evidence on file that confirms that Daniel and Andrew are May Bishop's children and that `May` and `Robert Tamou` legally adopted `Taina Shaun Tamou` and `Joval Margaret Tamou`. So we have data to support the inference that these people are a familial group.

While there are many complicated examples where incorrect inheritance can occur, such as a claim never being made at all or a person not being correctly reported as dead or the wrong type of inheritance being applied (there are 43 different types). This is a minimal example of a complex inheritance system.

Let us look at a fictitious example: suppose we want to find a person with the name `Sarah Wiki`. Lacking an immediate contact, one approach would be to try to find *the nearest relative for which we do have contact information*. A search in MLO for `Sarah Wiki` yields 11 ID's and 3 names, and MLO on its own can't tell us who is who. If we only have a person's name then we cannot tell if `Sarah Wiki` is the same person as `Sarah Wiki` or `Sarah Wiki`. We cannot tell if there are one, two or three individuals called `Sarah Wiki`. However if we know that the *real* `Sarah Wiki` has a brother called `Bob Wiki` then we can use this natural grouping to compare all of the `Sarah Wiki` records to see if either of them have a brother called `Bob Wiki`.

By filtering/blocking [39] our data, we can decrease the number of comparisons that need to be made. The complex relationships and inheritance rules in this problem may help rather than hinder as they allow us to create smaller units of interest. We can use minute book references from the MLO data set to infer sibling relationships on a block. If one could form a correspondence between the information in MLO and that in BDM, we could then say how many entities are at play, given their immediate family, and relate this back to the other names on the same land block back in MLO.

3.1 Data

3.1.1 Two sets of data

3.1.1.1 Māori Land Online

As the largest freely accessible rich data source connecting Māori people to their land, MLO is a unique repository of information about New Zealand's past and present. Its immediate value is as a legal repository (it is a list of who is an owner of which land block). But it is also, potentially at least, an enabler of social connection and holder of personal histories. This second value goes largely unrealised.

MLO can be considered reasonably accurate when it comes to blocks of land, yet reasonably inaccurate when concerning individual people. It is not meant to be about relationships *between people* but instead between a person and their land. In the data that is publicly ac-

Block ID :	17156
District :	Aotea
Title Order Type :	Partition Order
Title Order Ref :	59 TAR 97A
Title Notice Ref :	-
Title Order Date :	17/04/1953
Land Status :	Maori Freehold Land
Plan :	ML 1680
LINZ Ref :	397461
Area (ha) :	40.5733
Total Shares :	2160
Total Owners :	44

Soil Report (from Landcare Research)
Soil report for this block (open in new window).

Name	Trustee or Remainder Name	Alternate Names	Gend	Ownership Type	Min Bk Ref	Ratio	Shares	
(394083)								
Ratahi Tui (695444)			F	Absolute	010RGWG262		5.15465	
Ratahi Turei (695445)				Absolute	010RGWG262		5.15464	
Ratahi Whiro (339850)			M	Absolute	412 AOT 31-35		16.16098	[1]
Ratahi Whiro (394086)			M	Absolute			698.28102	
Robinson Andrew Clayton (945381)			M	Absolute	414 AOT 282-283		15.46406	[2]
Robinson Daniel Shayne (945380)			M	Absolute	414 AOT 282-283		15.46407	
Tamou Joval Margaret (945383)			F	Absolute	414 AOT 282-283		15.46406	
Tamou Taina Shaun (945382)			M	Absolute	414 AOT 282-283		15.46406	

Figure 3.1: **BLOCK : Ngatitara 26B.** This is an example of a MLO block, it is publicly available data from accessed from the MLO website on 10/07/2020 (*https : //www.maorilandonline.govt.nz/gis/title/*17156*.html*). This block is located in the Taranaki area, just north of Opunake. *(1 - blue)* Shows an example of two individuals on the same block. This, if we just looked at the name would seem to be a duplicate. However when looking at the number of shares that the two entries for `Ratahi Whiro` has, one has 16.16 and the other has 698.28. This is a huge disparity. In addition the `Ratahi Whiro` who has 16.16 shares also has an associated minute book reference, while the other one does not. This suggests that it is a coincidence that both men share the same name (although it is likely they are related), but that one is much older and perhaps even one of the original owners on this block. 698.28 is a large number of shares for one individual to own, much more than anyone else on this block. *(2- red)* Shows four individuals with the same share value (15.46) and the same minute book reference (414 AOT 282-283). In the classification adopted by the wider project, these individuals are all put into a single natural grouping [46]. The fact that they have different last names is not paramount, as we *know* that they have kept the same company. The data that we have available suggests that they all inherited their shares from the same person at the same court event.

cessible, time is latent variable, but it has been obscured: MLO conveys current ownership. We cannot see who a person inherited from or under what form of inheritance. There are many different types of inheritance through the Māori Land court. Inheritance is applied for through the Māori Land Court, but it is a complex process and not everyone who is eligible completes the process. In effect we are not able to see the real truth or what it should look like, only the current state of the system.

Because of the size of the MLO database, it is difficult for those who manage the land (like *PKW*) to know who should inherit. They make an attempt to follow up all the links that they have, however it is a hugely time intensive process. So it remains the responsibility of the individuals or the families to follow up and correct the land court data. This is an arduous process meaning that the MLO database becomes more and more out of date.

Other members of the Wai-te-ata Press team were able to extract natural groupings from MLO data [46]. These were based purely on the share block, the minute book reference and the share value. When individuals on a block share a the same minute book reference and share value we infer that they were at the same court hearing in regards to the piece of land. We group these individuals together as we know that they have been in each others company (at least once) and are likely to be siblings because of their same share value. This means that we have access to what can be thought of as sibling groups from MLO and that we can filter/block these records based on family surname. An example of how this might be applied can be seen in Figure 3.1.

3.1.1.2 Births, Deaths and Marriages

Births Deaths Marriages has natural groupings (families). BDM in contrast to MLO can be considered to be crisp and accurate when it comes to individuals (but does not contain anything to do with land blocks). It is a well maintained up to date database of all New Zealanders. Births Deaths and Marriages is maintained by the Department of Internal Affairs.

The individuals available in BDM records can be considered to be an almost perfect reflection of the truth in the eyes of the government. We know that an individual's legal name/identity does not always equate to a person's "real" identity but it is key piece of data for *PKW* to have in order to find someone. Dates are specific and correct, but there are no current locations. BDM captures events and moments where an individual changes.

For the purposes of this study we only have access to the Historic data (BDM-H), where the birth is more than 100 years ago. In our case we are interested in Births. The idea is that we can prove that the full data set would be of use to the project.

Other members of the Wai-te-ata Press team were able to identify sibling groups in BDM-H data "based on birth entries which shared the same surname and exact same parents' names" [46]. This means that we have access to 'natural groups' from BDM and that we can filter/block these records based on family surname.

3.1.2 Cenotaph records

Although MLO and BDM are the main focus of this research, a third data source was identified as potentially valuable by the Wai-te-ata Press team. There is a collection of Cenotaph records that has been maintained by the Auckland War Museum. This contains the details of New Zealanders that have served the country on active service since the time of the Māori land wars until the present day [69]. Individuals present in the Cenotaph records can (like BDM) be considered fairly crisp as these records have been actively maintained. They have also been updated by family members so that they accurately reflect ground truth. We

include the cenotaph data as an alternative data set, which is from around the same time period as the Historic BDM data. The raw cenotaph data was processed by other members of the Wai-te-ata Press team into a data set of brothers. Women were removed from this data set as last names were thought to be unreliable. This gives us reasonably crisp sibling groups that mimic the natural groupings we have in MLO. The major downside of this data set is that it does not contain any females.

3.1.3 So what?

Because we are able to generate sibling groups in both BDM and MLO, in theory we can use these groupings to find records for a person in MLO in BDM. Being able to identify an individual from MLO in BDM is allows us to deduce what happened to that person. We can tell if they are still alive, whether they got married or changed their name or who that person's living descendants are. In finding their BDM record we know their legal name and hopefully have a good chance at finding them. BDM is a rich source of leads for a *PKW* person to find shareholders. Because these sources are so different in what they provide, they provide complementary and potentially very valuable information to each other. The question is how to realise this value.

3.2 Problem set up

Having defined roughly the scientific and cultural landscape in which this project sits it is now important to introduce the particular problem that this thesis has focused on. It is a non trivial problem that cannot be solved by human eye at the scale of the real data.

An individual has:

- *A first name*. Generally all individuals have a first/given name however in some cases an individual may have no first name recorded in MLO.
- *A last name*. In some cases an individual and their siblings do not share the same last name - this may be through marriage or by choice. The concept of last names are a western introduction so not all individuals have one. We use last names as a method to filter and group individuals records. However once they are grouped we are actually not interested in the last names anymore as they provide no additional information. We know that the family groups we are comparing keep the same company. We are now most interested in first names as these (especially in a traditional Māori context) and so will focus on using first names to identify and link individuals.

It is worth noting that individuals also have ID numbers in the MLO data set however they have been shown to be untrustworthy so they are not used in the analysis.

First, some notation:

- $\mathbf{m} = [m_1, m_2, \ldots m_M]$ is a set of names (M in number, given names only), selected by a filtering process crafted to generate groups of *plausible siblings* in MLO data as per Section 3.1.1.1.
- m_{ij} is a character from an individual/name in $\mathbf{m}$.
- $\mathbf{b} = [b_1, b_2, \ldots b_B]$ is a set of elements (B in number, given names only) that corresponds to identities in BDM. By harvesting and filtering by surname we can be virtually certain that those in a given $\mathbf{b}$ are all siblings. This process is described in Section 3.1.1.2.
- b_{ij} is a character from an individual/name in $\mathbf{b}$.

- *Puzzle*. We define a *puzzle* to be a comparison of all of the **b** and **m** sets that contain individuals with a surname of interest. For example the `Lemon` *puzzle* contains **b** and **m** **groups** where all of the groups have at least one individual with the surname `Lemon`.

- $\mathcal{B}_A = [\mathbf{b}_1, \mathbf{b}_2, \ldots \mathbf{b}_B]$, B in number. All the BDM groups for a surname. This is the BDM part of the puzzle, where all $\mathbf{b}_j$ in $\mathcal{B}$ relate to the same surname, in this example A.

- $\mathcal{M}_A = [\mathbf{m}_1, \mathbf{m}_2, \ldots \mathbf{m}_B]$, M in number.All the MLO groups for a surname. This is the MLO part of the puzzle, where all $\mathbf{m}_i$ in $\mathcal{M}$ relate to the same surname, in this example A.

- *Linkages*. A linkage is an association between an **m** and a **b** set. We evaluate each linkage to determine its plausibility.

- *Alignment*. An alignment is an matching of individuals in **b** to individuals in **m**.

3.2.1 Test Data

We have access to both BDM-H and MLO data sets. However this is unlabeled data (i.e. we do not know if one group truely represents another). So instead we have to come up with our own test data.

To evaluate the ideas in the later chapters we need some notion of ground truth, which in general we don't have. We will use BDM-H data compared against itself to test if we are able to correctly link families and individuals, this allows us to imitate a "ground truth". We can take actual BDM-H b-groups from some set $\mathcal{B}_A$ (with last name A), add 'noise' to them in the form of new / omitted names and mis-spellings, and use them as proxy m-groups in a test. From these 'pretend' m-groups we can then try to distinguish between the (known) b-groups that led to them ($\mathbf{b} \in \mathcal{B}_A$), or a *different* set of b-groups, corresponding to a (randomly chosen) other surname ($\mathbf{b} \in \mathcal{B}_{\mathbf{A}'}$). We can evaluate whether our methods can distinguish between the two cases.

We therefore used a segment of the real BDM-H as a test set. This was duplicated used to represent both the BDM and the MLO set. This means that we created a situation where we know which **m** sets were created by which **b** sets. This allows us to assess how accurately the method is performing. Before being used in the test, some pre-processing was done and duplicates were removed. Each family group was processed so that stop words were removed and capital letters were changed to lower case. We shall call this the *ground truth data set* because it allows us to test our algorithms and methods *as if* we knew ground truth.

A data set of 80 family names were used. These *puzzles* were selected from MLO because they appeared between 10 and 20 times and they are large enough to be interesting to compare. There are 892 **m** sets from those 80 names and 3568 individuals in this test set.

These are all of the last names from the following data set.

```
["Amundsen", "Armishaw", "Baron", "Begley", "Benbow", "Bracken",
"Bretherton", "Bruning", "Byford", "Castles", "Clendon", "Conaghan",
"Coogan",  "Cording", "Crabbe", "Deadman", "Dowie", "Faithfull", "Flintoff",
"Grayson", "Greeks", "Guildford","Gutsell", "Hardcastle", "Harkins",
"Hasler", "Heaven", "Hennessey", "Hoey", "Hollamby","Hornby", "Jepsen",
"Kilkelly", "Loughlin", "MacKey", "McGlone", "Meager", "Nissen", "Nunn",
"Pook", "Rendell", "Saxton", "Scrivener", "Sharples", "Shepard", "Singer",
"Skipworth", "Smillie", "Snowdon", "Stainton", "Steen", "Symon", "Triggs",
"Trow", "Twist", "Wallbank", "Wards", "Wesley", "Woon", "Wylde"]
```

Interestingly there are no obviously Māori surnames in this set. There are Māori first names associated with these last names. It is worth noting that prior to Pakeha arrival in New Zealand many Māori did not use last names. In colonial times and since, many Māori adopted a last name. Often these were a transliteration of part of their Māori name. An example of a transliteration is using `Wilson` in place of `Wirihana` [53].

3.2.2 Noise

In order to simulate the poor data quality of MLO, noise was added to the fake **m** set described above. This section describes the noise model that is used to "rough up" the test data. There are a series of parameters which can be changed to turn up and down the probability that a family member is left out entirely, a name is left out or that a word is miss-spelt. It is difficult to pinpoint the exact values that these parameters should be set to as we have no ground truth to start with. We use a first order Markov model trained on a reasonable subset of BDM-H to generate the misspellings.

We set the noise value later in the thesis. The noise value is used to set the following things:

- The probability that an individual in a family is completely removed from the data set.
- The probability that a new individual is added to a family.
- The probability that an individual letter in a person's name is changed (mis-spelled).

For example when noise level is set to 0.2, this means that each letter of each name has a 0.2 chance of getting changed to another letter and each word has a 0.2 chance of being removed or joined with another word. There is also a 0.2 chance of having an additional word added. Here are a few examples of names before and after going through the noise model:

`Ralph Morton` $\rightarrow$ `Razph Moqtoo`

`Arnold James` $\rightarrow$ `Armol_ Jam-s`

`Harold Oscar` $\rightarrow$ `Osceht`

`Frank Hector` $\rightarrow$ `FganubHectdr`

`Mervyn` $\rightarrow$ `Merhyn`

`Laurie Howard` $\rightarrow$ `Laxrie Hoosrd`

`Reginald` $\rightarrow$ `Regiotld William`

The names become much harder to decipher when the noise model is applied. We can observe various misspellings, additions and removals in the example. It no longer looks like the original, and whilst it looks pretty bad we want our models to be robust to really corrupted data.

3.2.3 Finding the missing

Consider the question of the origin of a particular family/set $\mathbf{m}$ derived from MLO. The question "Who are they, really?" becomes "Which $\mathbf{b}$ sets are most likely to contain the true identities of people in $\mathbf{m}$?". Notice we are approaching this as an inference problem over *groups* as opposed to individual identities. Obviously the identities of individuals will play a role in this. One of the interesting questions is to what extent the best alignment will suffice in inferring the best $\mathbf{b} \in \mathcal{B}_{\mathbf{m}}$.

A task then: given a particular $\mathbf{m}$, and a set $\mathcal{B}_{\mathbf{m}}$, we would like to assess the relative plausibility of each $\mathbf{b} \in \mathcal{B}_{\mathbf{m}}$ as being the group of people behind the names in $\mathbf{m}$. From a Bayesian perspective, this is the posterior probability $P(\mathbf{b} \mid \mathbf{m})$. So what does that calculation involve?

By Bayes' theorem,

$$P(\mathbf{b} \mid \mathbf{m}) = \frac{P(\mathbf{m} \mid \mathbf{b})\ P(\mathbf{b})}{P(\mathbf{m})} \tag{3.1}$$

Where the denominator is not important

$$\propto P(\mathbf{m} \mid \mathbf{b})\ P(\mathbf{b}) \tag{3.2}$$

One could argue that the prior over $\mathbf{b}$ in absence of any other information is flat, and so

$$P(\mathbf{b} \mid \mathbf{m}) \propto P(\mathbf{m} \mid \mathbf{b}) \tag{3.3}$$

That is, to evaluate beliefs about $\mathbf{b}$ given $\mathbf{m}$, we should look to the "forward" probability of $\mathbf{m}$ given $\mathbf{b}$, which is something we can build a concrete model of, it seems.

3.2.4 How do we best calculate P($\mathbf{b} \mid \mathbf{m}$)?

It is tempting to think about matching one group with another group by aligning each individual and assessing the number of acceptable matches. If we were confident that the data was perfect and complete, meaning that every individual in the group $\mathbf{b}$ was in $\mathbf{m}$ then this is a good strategy because we are able to assess the group as a result of assessing the individuals. Note that was are not taking into consideration any natural groupings or structure within the groups. However with messy data (like MLO) we do not believe that every individual in group $\mathbf{b}$ is is $\mathbf{m}$. This means that we must rely more on the structure and the natural groupings within our data.

In Chapters 4, 5 and 6 we focus on linking groups together using natural grouping techniques. In Chapter 4 we discuss methods of linking groups together (linkage methods) using an aggregation of names and the text analysis method of Ngrams. We also introduce several ways to reject linkages in Chapter 5, it is important to note that some of these apply in Chapters 7 and 8 as well.

In Chapters 7 and 8 we discuss the alternative approach where groups are linked together using specific individuals. This is done by assessing name similarity using edit distance or Ngrams and calculating an alignment between groups.

Chapter 4

Group to group linkage methods

We are looking for a family in **b** that is likely to match a family in m because they have similar names and similar structures. This chapter investigates methods of assessing linkages between **b** groups and **m** groups without addressing the question of any specific alignment. We propose several methods that link **b** sets to **m** sets.

4.1 The simplest model possible - MATCH

The simplest model finds names in **b** that exactly match names in **m**. Let us call this exact matching algorithm MATCH. We calculate a score from counts of exact matches which when normalised gives us $P(\mathbf{m}|\mathbf{b})$. The proposed model aims to strengthen linkages where there is an identical name match, i.e. the same name appears in both the **b** and the **m** group. This is a simple approach and we accept that failures will occur when we introduce any kind of noise, be that spelling mistakes or nicknames. However we use this as a straw man in order to propose more robust models.

In this method we normalise the number of exact name matches between **m** and **b**.

$$Score = \sum_{x \in M, y \in B} (1 - \delta_{x,y})$$

We allow duplicates in both B and M. We simply check for each name in M if the string appears in B. This means that if there are two instances of the same string in M, the score is the same. We are only interested in first names as last names are used to identify the group.

4.1.1 A specific example of exact string matching

Let us look specifically at how the MATCH model works on a real family. We are using our test *ground truth data set* described in 3.2.1, here displaying just the results from the family with the surname `Lemon`. We can see the results of this on the left in Figure 4.1. Since we are comparing two `Lemon` family sets that are exactly the same, we know the 'true' result and can measure success by the strength of the diagonal. There is a very distinct diagonal of the heat map.

We then also look at the true `Lemon` family's MLO data to see what the performance is like. This is displayed on the right in Figure 4.1. It is, as previously discussed, it is impossible to tell the true accuracy of the comparison with real a real MLO to BDM-H data because we do not have labelled data and do not have access to the actual ground truth. However when the "winning" results were inspected there was very little evidence that the matches were anything but an example of names appearing in the same sets at random. This was

Figure 4.1: **Heat maps showing comparisons between `Lemon` b and m groups.** Each row corresponds to a **b** group (essentially, a family) identified in the BDM data. Columns correspond to **m** groups, i.e. lists of *possible* siblings, these are from different land blocks (there could be overlap as a person can be an owner of more than one block). These two heat maps show the un-normalised counts i.e. raw counts of identical matches between the two families. The colour of each square, indicates the number of identical name matches between the **b** and **m** families. Darker colours represent higher counts. `Left`: The `Lemon` family, with no last names. This is using our ground truth test data where BDM is compared against BDM. A strong diagonal band in this figure indicates that on clean test data we can (as we would expect) identify the true source of each family. *Right*: The same test but on real MLO-BDM data, on the `Lemons` family. There is no strong diagonal and we cannot know without checking every square which ones are real links.

especially evident in the larger families where there was a higher likelihood that they contain a common name. An example of one such family is the one at index 0 on the x axis in 4.1 where many green squares are present.

Problems with the MATCH model:

- It does not include alignment in any way. This means that we are not able to specifically say which person is which, only that we suspect the two natural groupings contain the same individuals.
- It is not robust to noise. This means that partial matches such as spelling mistakes or nicknames are not counted.

4.2 Markov models for text - NGRAMS

An alternative way of assessing similarity between natural groupings without looking for exact matches is to use a Markov model for text to identify the likelihood of **b** creating **m**. Let us call this method NGRAMS going forward.

By the product rule we can write the probability of a string by the product of each character in the sequence conditioned on those that come before it as follows (where letters in `tt type` indicate characters rather than variable names):

$$P(\texttt{cat}) = P(\texttt{c})\ P(\texttt{a}|\texttt{c})\ P(\texttt{t}|\texttt{ca})$$

and so on. This cannot go on indefinitely as we do not have any good model of $P(\texttt{y}|\texttt{catter})$. Markov models built on Ngrams truncate the prefix to some integer n which we set.

The simplest of these is a bag-of-words generative model in which we truncate the whole string and set n to be zero. We can approximate the probability of a string **S** by the product

of each character:

$$P(\mathbf{S}) = \prod_i P(s_i) \tag{4.1}$$

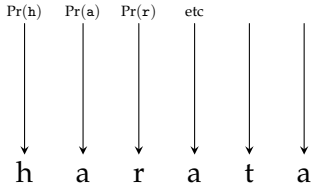
where s_i is the i^{th} character in $\mathbf{S}$ and $P(\mathbf{S})$ is the overall probability of the character s in some background corpus of text (like MLO), so that

$$P(\texttt{cat}) = \Pr(\texttt{c}) \Pr(\texttt{a}) \Pr(\texttt{t})$$

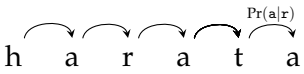
In log space this becomes additive:

$$\log P(\mathbf{S}) = \sum_{i=1}^{L} \log P(s_i) \tag{4.2}$$

Graphically, for the name harata:



In a first-order Markov model we truncate to a prefix of a single character, so that we choose our next letter based on the previous one:

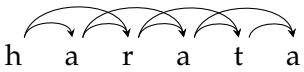


The log likelihood is:

$$\log P(\mathbf{S}) = \log \Pr(s_1) \; + \; \sum_{i=2}^{L} \log \Pr(s_i \mid s_{i-1}) \tag{4.3}$$

This bigrams model captures text structure at the pairs-of-letters level, but is blind to longer range dependencies.

One can easily extend to a model based on *trigrams* (i.e. a 2nd order Markov model):



For which the log likelihood is:

$$\log P(\mathbf{S}) = \log \Pr(s_1) \; + \; \log \Pr(s_2 \mid s_1) \; + \; \sum_{i=3}^{L} \log \Pr(s_i \mid s_{i-2}, s_{i-1}) \tag{4.4}$$

and so on. The higher the order, the more long-range structure is able to play a role, but also the higher the complexity of the model, with the attendant risk of over-fitting. We want a model that generalises, but is simple to explain. Later on in this chapter we will experiment over different values of n to set the size of the truncation.

As a general form,

$$\log P(\mathbf{S}) = \sum_{i=1}^{L} \log \Pr(s_i \mid s_{\texttt{pre}_i}) \tag{4.5}$$

where pre is the previous k characters for a k^{th} order Markov model .

Regardless of the order of Markov model chosen, if the background corpus on which it is based is $\mathcal{D}$ we could denote the per-character probability as $P_{\mathcal{D}}(s|\texttt{pre})$.

And if you were to build it based on a single string, say str, denote that $P_{\texttt{str}}(s|\texttt{pre})$.

The strengths of this is that the model can calculate the probability of strings of any length, and the approximation it makes is simple and explicit. Note $P(m)$ is automatically normalised:

$$\sum_{\text{all } m} P(m) = 1$$

A model for names that change

Our score is the log likelihood for string b to give rise to string m. To be plausible as a model of name change, we want this to be strongly tied to b itself of course, but to *also* admit the possibility of substantial additions or deletions: we want $P(m)$ to be a long tailed distribution.

We proposed a mixture of two distributions, the NGRAM distribution made from elements in **b** and the NGRAM distribution made from elements in BDM. For each character s_i in **S**:

$$P(\mathbf{S}) = \beta\, P_b(s_i) \;+\; (1-\beta)\, P_{\mathcal{D}}(\mathbf{s}) \tag{4.6}$$

with β a value that needs to be tuned/set. The model is that with β% chance the next letter is based on the statistics of the key name (which is b), and (1-β)% chance that it is from the background distribution of m-names in general. We will test for appropriate values of β later in this chapter.

NGRAMS gives us a model for how a name (or several names) in **b** might give rise to a single name **m**. This lets us find a natural way to evaluate a linkage between a **b**-set and an **m**-set: a *score* $\log P(\mathbf{m}|\mathbf{b})$ for a particular pairing. This allows us to evaluate $\log P(\mathbf{m}|\mathbf{b})$, either by averaging or optimizing the alignment, and hence let us compare different **b**-sets as hypotheses for an **m**-set.

However it also suggests a very direct way to evaluate $\log P(\mathbf{m}|\mathbf{b})$ without the need to look find any alignments: We use substrings from **b** to generate substrings in **m**.

$$m_i = name \in \mathbf{m} \tag{4.7}$$

$P_{\mathcal{B}}$ denotes the NGRAMS model made from names in **b**

$P_{\mathcal{D}}$ denotes the NGRAMS model made from names in BDM

We can find the log probability under the same β mixture above :

$$\log P(m_i \mid b_i) = \sum_{i=1}^{|\mathbf{m}|} \log P(s_i \mid s_{\texttt{pre}_i}) \tag{4.8}$$

where s_i are the characters in m_i, with probabilities given by the mixture model:

$$P(s_i \mid s_{\texttt{pre}_i}) = \beta\, P_{\mathcal{B}(s_i|s_{\texttt{pre}_i})} \;+\; (1-\beta)\; P_{\mathcal{D}}(s_i \mid s_{\texttt{pre}_i}) \tag{4.9}$$

Note that $P_{\mathcal{B}}$ is built from the strings of the **b** set. This means that there is no alignment-specific content in this probability. The advantage of this is that it is a direct and highly accessible way to score many sets of names. And the calculation is highly tractable. The disadvantage is that there is no alignment information built into the score and two families that are different but have the same n-grams distribution will score the same. For example the **b**-set of `[Wiremu, Piri, Atareta]` is destined to get the same score as `[Reta, Wiri, Pemu, Ata]`. And that equivalence will be true *regardless of* **m**.

How to calculate scores, $\log P(\mathbf{m}|\mathbf{b})$

We can think of feeding an automaton the string $\hat{m}$ as a stream of characters, with the output being a float: the log probability of that string.

- For each **b**, build the ngram distribution for **b**. It computes and stores a dictionary for $\log P(s_i \mid s_{\texttt{pre}_i})$, as per Equation 4.9. It is able to ingest a string of characters and output a float (as per Equation 4.8, Which is simple as it is just a sum of the appropriate values).
- For each **m**, attempt to make m_i given the Markov model $P_{\mathcal{B}}$, giving score S.

So for each **m**-set we've now got the top scoring **b**-sets and their scores. At the moment each of the scores is in log space so to find $P(\mathbf{m}|\mathbf{b})$, we have to take the exponential of what we have. We also need to normalise the scores to get the posterior probability under this model.

4.3 Smoothing

The problem is that many of our **m** family NGRAM distributions contain combinations that the **b** family distribution (and the base distribution) do not have. If, for example the **m** set has a sub string `zzql` that it has simply never seen before in our base distribution or in the **b** dictionary we have to be able to handle it as we cannot return zero for $P(\texttt{l}|\texttt{zzq})$. Returning zero is bad as the probability of the ngram `zzql` occurring is obviously very low but it is not actually never going to occur. Since we multiply the probabilities of all of the NGRAM events together, one zero would lead to failure of the whole process. Adding some form of smoothing protects us from this [57].

4.3.1 Additive smoothing

In additive smoothing a small parameter is added to all categories (even ones we have not seen yet). This is so that when an unseen combination is found in **m**, we do not have to return zero. The most common versions of additive smoothing are Laplace and Jefferies [57]. In Laplace a count of 1 is added to the frequencies of each category respectively, for Jefferies the same principal is used except a count of $1/2$ is used. We are using a very low smoothing coefficient in place of this count, where $\epsilon = 0.001$. This returned instead of zero when the probability of a string/sequence is not known. We are aiming to find families where the distributions are very similar between the **m** and **b** sets.

The problem with this approximation is that in cases where there is a lot of unknown values the $P(\mathbf{b}|\mathbf{m})$ is driven right down to pretty much zero.

4.3.2 Backoff

Backoff smoothing is a more sophisticated alternative to additive smoothing. Murphy [59] suggests that the solution to the problem with additive smoothing to use backoff smoothing. This is where instead of using a fixed n, n is varied to find a distribution where we have higher confidence in having seen that combination before [58].

Stupid Backoff first implemented by Google in 2007 [62] was chosen because it performs almost as well as traditional backoff smoothing methods such as Katz [60] or Kneser-Ney smoothing [61] but is much faster and simpler. In Stupid Backoff you check for the occurrence of your n-length string, if it has been seen before you return the probability of that even occurring. If there has never been an occurrence of that substring before then n is temporarily reduced by 1. The search is then undertaken at the new value of n, n being reduced until an occurrence is found. Each time the n value is reduced a penalty term α is multiplied to the final event probability, this represents the fact that we are less and less sure about the word overall as n decreases.

For example: if we have the name `Ssarha` and $n = 4$, we would first look for `Ssar`, If we have never seen `Ssar` before then we reduce to $n = 3$ look `sar`, then $n = 2$ `ar` and finally $n = 1$ `r`.

$$S(w_i|w_{i-k+1}^{i-1}) = \begin{cases} \dfrac{f(w_i|w_{i-k+1}^{i})}{(w_i|w_{i-k+1}^{i-1})} & \text{if } f(w_i|w_{i-k+1}^{i}) > 0 \\ \alpha S(w_i|w_{i-k+2}^{i-1}) & \text{otherwise} \end{cases}$$

This is slightly different from traditional backoff methods in which frequency tables are used. In these other methods [61] [60] if the string has been seen before but frequency of the event is too low, backoff occurs anyway. Stupid Backoff is slightly faster because the frequency tables are not kept and backoff occurs only when the string has never been seen before. The creators of Stupid Backoff Brants et al, commented that "The name originated at a time when we thought that such a simple scheme cannot possibly be good. Our view of the scheme changed, but the name stuck"[62].

To compare Stupid Backoff against approximated additive smoothing we compared the log likelihoods of the two methods on a representative sample of MLO. If the log likelihood is higher this means that the method is performing better.

method	n=1	n=2	n=3	n=4	n=5	n=6	n=7	n=8
Additive	−6 571.732	−450.507	−187.922	−115.842	−83.733	−69.203	−59.893	−53.207
Backoff	−6 571.732	−319.701	−53.800	−24.051	−17.503	−15.706	−15.161	−15.120

Table 4.1: **Log likelihoods of (Stupid) Backoff and Additive (approximation) smoothing**. This table shows the differences between Stupid Backoff and an approximation comparison. There is a clear divergence in the log likelihoods as the n value increases, larger values are better as they are produced when the model is a better predictor.

Table 4.1 shows the Stupid backoff outperforming the additive smoothing method showing a higher value. As n increases, the log likelihood of stupid back off is higher than the log likelihood of the approximated additive smoothing at the same value of n. When n equals 1, there is no difference between the two methods. This is because both are effectively doing the same thing, as the backoff model has no smaller n to backoff to. Both methods use the same first order Markov model to produce the same result and have the same log likelihood. At larger values of n, the stupid backoff model backs off when it cannot find the substring it is looking for, and looks at a smaller n value. The additive model instead returns a very

small constant. As n increases the more likely the additive model is to return only the small constant and not a real probability. Therefore performance of the additive model decreases as n increases, because it returns a value further and further from the truth.

4.4 Finding β and n

A series of tests and experiments were carried out to test the performance of this simplified ngrams model. The group-to-group method uses the MATCH and NGRAMS methods as described earlier to evaluate $P(\mathbf{m}|\mathbf{b})$ and compare different **b**-sets as hypotheses for an **m**-set. It is important to set a few of the threshold variables. The following experiments were designed to test the performance of several different variables as well as the model itself.

We need to find an appropriate value of β, this is the mixing coefficient described in Section 4.2. β sets the mixture of the distribution of the ngram frequencies from the **b** distribution and the NGRAM frequencies base distribution made from a large partion of BDM-H ($P_{\mathcal{B}}$ and $P_{\mathcal{D}}$). A range of different β values were tested, from 0.0 to 1.0 at intervals of 0.1. We also want to know how the different values of n perform, where n is the length of the substring used by the NGRAMS method. n values from 1 to 8 were used. Base distributions of n larger than 8 were too slow to calculate, store and load. We want to simulate a real scenario, where siblings are not listed in MLO in the same order as they are in BDM. We tested the algorithm against a data set where the family members within a subgroup are reordered, to make sure that the order was not important.

These are aggregated results over the 446 families and 60 surnames from the test data described in Section 3.2.1. A True Positive result is considered to have occurred when the real family has the highest score. The performance of the methods on noisy data is most important as we believe this to be the most like a realistic scenario. We cannot assess the True Negatives without rejection. This is added in the next chapter.

4.4.1 MATCH performance

Noise	True +	False -	True -	False +	Precision	Accuracy	f1
0.000	**0.857**	0.143	0.571	0.429	0.667	0.714	0.750
0.11	0.571	0.429	0.571	0.429	0.571	0.571	0.571
0.2	**0.571**	0.429	0.286	0.714	0.444	0.429	0.500
0.3	0.286	0.714	0.429	0.571	0.333	0.357	0.308
0.4	0.143	0.857	0.143	0.857	0.143	0.143	0.143
0.5	0.143	0.857	0.143	0.857	0.143	0.143	0.143
0.6	0.143	0.857	0.143	0.857	0.143	0.143	0.143
0.7	0.143	0.857	0.143	0.857	0.143	0.143	0.143
0.8	0.000	1.000	0.000	1.000	0.000	0.000	0.000
0.9	0.000	1.000	0.000	1.000	0.000	0.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	0.000	0.000

Table 4.2: **Accuracy of the MATCH linkage method on test data.** The accuracy of the algorithm significantly decreases when the noise is increased. At zero noise the true positive rate was high, correctly linking the right **b** set to the **m** set in 85.7% of **m** sets. When the level of noise increased to even 0.1 the true positive performance decreased to just 57.1%. When the noise rate was increased to more than 0.7, there were no true positive results recorded.

Table 4.2 shows the linkage accuracy of the very simple MATCH algorithm at a variety of noise levels. Noise is applied using the noise model is discussed in section 3.2.2. This is the

expected behaviour of the MATCH model, when noise is not present it is a simple and trivial task to do exact name matches. Because we know that MLO is a noisy data set the poor performance on noisy data shows precisely why more complex models and algorithms are needed. This means that we would expect a high volume of noise and a simple name match would not be sufficient.

4.4.2 NGRAMS model

β	n=1	n=2	n=3	n=4	n=5	n=6	n=7	n=8
0	0.137	0.137	0.137	0.137	0.137	0.137	0.137	0.137
0.1	1.000	1.000	1.000	1.000	1.000	0.995	0.986	0.974
0.2	1.000	1.000	1.000	1.000	1.000	0.997	0.991	0.985
0.3	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.996
0.4	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.5	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.6	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.7	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.8	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.9	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Table 4.3: **NGRAMS linkage performance with no noise.** This table shows that the accuracy across all n values is very good even at low values of β.

Table 4.3 shows the performance of the NGRAMS model over different values of β and n. By varying β we are using different mixtures of the two distributions $P_{\mathcal{B}}$ and $P_{\mathcal{D}}$. This is the mix between the NGRAM probabilities that are drawn from the base distribution and the NGRAM probabilities that are drawn from the names in the target **b** set. In this test the performance of the model is poorer at the larger values of n. The accuracy was high until n=6, where the accuracy drops off. The β value did not affect the results of the algorithm much, except for when β was set to zero - i.e. when the mixture is mostly/entirely drawn from the base distribution. That is as expected because all of the linkages contain exact matches in our test data and we have applied no noise to this experiment.

If the parameters are set to have $n = 1$ and $\beta = 0.0$ (top row). This is because the model has no information about the letters in each family and the results are drawn entirely from the base distribution.

Figure 4.2 shows the behaviour of the model when the parameters are set to have n=3 and β=0.1. The brighter coloured squares represent an linkage that the model predicts to be likely. Here you can see a diagonal forming, from bottom left to top right. Figure 4.3 shows the results with increased β and introduced noise. There is a diagonal but it is not as clear as in Figure 4.2. There is now sprinkling of incorrect predictions outside the diagonal.

4.4.2.1 NGRAMS model with noise introduced

By introducing noise in our test data we are able to test if our NGRAMS model are fragile or robust. We will use the same settings as before and assess the accuracy of the

Table 4.4 shows that the accuracy across all n values is very good even at low values of β. The introduced noise does lower the performance this suggests that the NGRAMS is robust to noise.

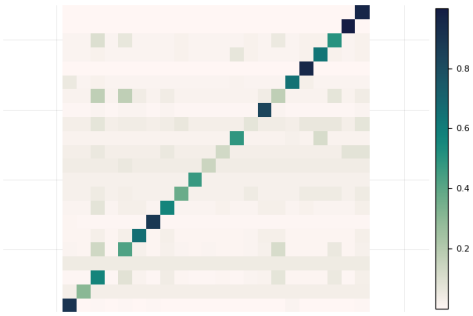


Figure 4.2: Scores for each family, where n=4 and β = 0.2, noise = 0

Figure 4.3: Results for each family, where n=6, β = 0.9, noise = 0.2

β	n=1	n=2	n=3	n=4	n=5	n=6	n=7	n=8
0.000	0.137	0.137	0.137	0.137	0.137	0.137	0.137	0.137
0.1	0.997	0.985	0.979	0.957	0.944	0.910	0.876	0.903
0.2	1.000	0.977	0.972	0.975	0.958	0.928	0.936	0.924
0.3	0.995	0.996	0.978	0.978	0.973	0.944	0.941	0.927
0.4	0.997	0.985	0.977	0.969	0.966	0.947	0.944	0.935
0.5	0.997	0.997	0.989	0.984	0.986	0.959	0.959	0.939
0.6	0.992	0.994	0.986	0.984	0.976	0.958	0.963	0.939
0.7	0.998	0.992	0.987	0.981	0.966	0.935	0.942	0.908
0.8	0.994	0.981	0.977	0.973	0.951	0.947	0.918	0.923
0.9	0.996	0.985	0.985	0.966	0.951	0.935	0.917	0.929
1.0	0.985	0.980	0.966	0.945	0.905	0.893	0.882	0.830

Table 4.4: **True positive accuracy on test data using the NGRAMS linkage method with noise = 0.2.** Accuracy was very high across all values of $\beta and n and best at n=1 and \beta = 0.6$. This is a change from when there was no miss-spellings where the performance was fairly even across all of the values of n but a high β was important (likely due to the lack of introduced spelling errors). Accuracy's at the larger β values is also poor, this can be explained by the fact that at $\beta = 1$ the distribution is solely drawn from the **b** set distribution.

4.4.3 Values for β and n

Because True Positive linkage accuracy was very good across all β and n values (except for $\beta = 0$) we have chosen to use $\beta = 0.5$ and $n = 4$ for further experiments. Whilst all values of β show good performance, having an even mixture of $P_{\mathcal{B}}$ and $P_{\mathcal{D}}$ will prevent over fitting to the **b** set.

Chapter 5

Rejection Methods

In order to asses the performance of the group-to-group linkage algorithms from Chapter 4 we developed several rejection methods which accept or reject the candidate linkage found in Sections 4.1 and 4.2. We need to determine if any of the chosen families are likely to be a real match or just the best performing comparison from a set of low probability families. The following methods evaluate the results distributions for each family and determine which linkages we should reject.

Three methods are assessed are:

- A threshold defined by the worst result, when the names in a family are compared against itself (AL1).
- An entropy assessment which uses the property of surprise or uncertainty to distinguish real matches from poor ones (ENTROPY).
- A threshold defined by the ability of $P_{\mathcal{D}}$ to generate $\mathbf{m}$ compared to $P_{\mathcal{B}}$ (BTN).

5.1 Rejection setup

5.1.1 Assessing Classifications and Correctness

After the linkage methods have been applied to the data set, and records have been linked or not linked to other records, an assessment needs to be done on the correctness of these links. In principle this is only possible to calculate on labeled data. In a traditional classification problem there are four different possible classifications that could be made.

- *True Positive* - this is defined as two groups of records correctly matched by the algorithm, i.e. two groups that relate to the same ground truth family.
- *False Positive* - This is defined as two groups of records that are matched by the algorithm but actually refer to two different ground truth families.
- *True Negative* - this is defined as two groups of records that were correctly not matched by the algorithm and relate to two separate ground truth families.
- *False Negative* - this is defined as two groups of records that are incorrectly not matched by the algorithm and actually do relate to the same ground truth family.

		prediction outcome	
		p	**n**
actual value	**p′**	True positive (TP)	False negative (FN)
	n′	False positive (FP)	True negative (TN)

Figure 5.1: **Confusion Matrix.** This figure shows the different confusion values depending on the predicted outcome and actual value.

These four classifications can be seen in Figure 5.1.1 which shows a visual definition comparing actual value to predicted value.

The number of true linkages (P) is the sum of the correctly linked records and the false negatives: $P = TP + FN$. The number of linkages that the algorithm estimates (E) is the sum of the correctly linked records and the incorrectly linked records: $E = TP + FP$

These can be looked as a rate of correctness: the False positive rate and the False negative rate are usually used in assessment of classification models or algorithms.

$$FNR = \frac{FN}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

We have used the additional measures of accuracy, precision and the F1 score to assess the performance at various threshold levels. Accuracy is a measure of truth and answers the question of ”how good a predictor is the algorithm”. Low accuracy occurs when real links are not found and when false links are wrongly thought to be real.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision is a measure of variability.

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP + FP}}$$

The results of a set of data can be said to be accurate if the average result is close to the true value of the quantity being measured. In our case accuracy occurs when true linkages are classified as being true and false linkages are classified as being false. Results can be said to be precise if the values being measured are close to each other.

F1 is a the harmonic mean of the precision and recall (recall being the fraction of true links correctly classified), which is a commonly used measure for natural language processing applications [65].

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

5.1.2 Rejection Data

In order to test the effectiveness of our methods we need to add suitable negative cases to our existing test data set (described in section 3.2.1). The rejection test data needs to be very similar to the real set, but without the true family. We took the true test data from Section 3.2.1 and for each family comparison, replaced the "true" family with a random family. The new random family is generated randomly from names in the BDM-H test set. First, by selecting a random number of family members between 2 and 10 and then randomly selecting this number of first names from BDM-H names. For example in this fictional comparison:

$$\mathbf{b} = [\texttt{alice}, \texttt{jane}, \texttt{wiremu}], [\texttt{james}, \texttt{john}, \texttt{kate}], [\texttt{henry}, \texttt{mathew}, \texttt{valerie}]$$

The family of interest $[\texttt{alice}, \texttt{jane}, \texttt{wiremu}]$, is replaced by $[\texttt{ariana}, \texttt{geoff}, \texttt{tama}]$. So $\mathbf{b}$ becomes

$$\mathbf{b^*} = [\texttt{ariana}, \texttt{geoff}, \texttt{tama}], [\texttt{james}, \texttt{john}, \texttt{kate}], [\texttt{henry}, \texttt{mathew}, \texttt{valerie}]$$

.

It is important to have good test rejection data so that we have evidence our rejection methods can reject real looking negative linkages as well as obvious mistakes. This is why the replacement families must be generated by BDM training data. A poor choice of replacement in the example above would be $[\texttt{zzzzz}, \texttt{mmmm}, \texttt{3tgs2}]$ because it would never ever appear in the training data. This gives us confidence that our results are reasonable and our methods could be used on real data.

5.2 At-Least-1 - AL1

When an employee of PKW looks at two lists of shareholders they assess similarity by checking for individuals who appear in both lists . As humans we can identify matches even when there are partial matches i.e. spelling mistakes or nicknames. We would not think that two groups of people had any chance of being linked unless there was *at least* one match. Lets look at a "test case" in which $\mathbf{b} = [\texttt{Thomas}, \texttt{Steve}, \texttt{Bob}]$ and $\mathbf{m} = [\texttt{Stephen}, \texttt{Thomas}, \texttt{Jerry}]$. If you simply count the exact matches, you get a score of 1 for the theory that $\mathbf{b}$ made $\mathbf{m}$:

$$[\texttt{Thomas}, \texttt{Steve}, \texttt{Bob}] \rightarrow [\texttt{Stephen}, \texttt{Thomas}, \texttt{Jerry}]$$

This linkage is at least a candidate, and as it's got a perfect match between `Thomas` and `Thomas`. It seems like it should be more convincing than, say, an empty list making $\mathbf{m}$:

$$[] \rightarrow [\texttt{Stephen}, \texttt{Thomas}, \texttt{Jerry}]$$

But the test case also has a second, but partial, match (`Steve`). Maybe one match is suspicious, but intuitively two starts to get quite convincing. So instead of only counting exact matches, we would rather allow partial matches (like `Steve - Stephen`) to play a role too. So if we were to use the naive approach where

$$P(M|B) = \prod_i P(m_i|B)$$

(with i counting the elements of $\mathbf{m}$) means we get

$$\frac{P(\mathbf{m}|\mathbf{b})}{P(\mathbf{m}|\phi)} \approx \frac{\sim 1 \times 1 \times 0}{\epsilon^3}$$

where ϵ is the rough probability of a name under the background distribution (which will typically be pretty small, but not microscopic).

The almost-zero on the numerator (probability of `Jerry` under the `Thomas` and `Steve` counts) acts like a "veto", killing that probability. This model does not favour **b** over ϕ: so this linkage is not a candidate.

This is where the property of the NGRAMS method becomes useful, with the `Steve - Stephen` match, there is a 3 letter overlap. So where $n < 4$ there should be "points awarded" to this match.

We propose the use of a probabilistic threshold value of $P(\mathbf{m}|\mathbf{b})$ that is set to be as good as a single perfect match. This makes use of the benefits of the NGRAMS methods, that allow for partial word matches and misspellings. We want this method to keep family comparisons who do *at least* as well as a match with one name exactly correct. This could be in the form of two partial matches or six tiny partial matches where n=2 , but we want there to be some resemblance.

5.2.1 How to reject with AL1

For a linkage between an $\mathbf{m} = [m_1, m_2, \ldots m_M]$ and a $\mathbf{b} = [b_1, b_2, \ldots b_M]$, calculate

$$O = \{s|s \text{ is the} P(\mathbf{m}|\mathbf{b}), \text{ there } \mathbf{b} = [m_i]\}$$

Each element in O is the $P(\mathbf{m}|\mathbf{b})$ where $\mathbf{b} = [m_i]$ for each $m_i \in \mathbf{m}$. We will reject a linkage if the minimum of the :

$$Classification = \begin{cases} accept & \text{if P}(\mathbf{m} \mid \mathbf{b}) \geq \min O \\ reject & \text{otherwise} \end{cases} \tag{5.1}$$

Let us look at this using our example from earlier with `Thomas`,`Steve` and `Bob`. Because:

$$[\texttt{Thomas}, \texttt{Steve}, \texttt{Bob}] \rightarrow [\texttt{Alice}, \texttt{Jane}, \texttt{Wiremu}]$$

Has a lower score than the worst score from:

$$[\texttt{Thomas}, \texttt{Steve}, \texttt{Bob}] \rightarrow [\texttt{Thomas}]$$

$$[\texttt{Thomas}, \texttt{Steve}, \texttt{Bob}] \rightarrow [\texttt{Steve}]$$

$$[\texttt{Thomas}, \texttt{Steve}, \texttt{Bob}] \rightarrow [\texttt{Bob}]$$

Then we would reject [`Alice, Jane, Wiremu`]. But we would not reject a set [`Alice, Bob, Wiremu`] as there is one exact match and it therefore would produce a higher score.

5.3 A threshold using ENTROPY

We can make use of the Shannon entropy of the posterior distribution $P(b|m)$, which reflects how much the distribution is spread out over the available families. Low entropy points to a strong winner or winners, and hence we might reasonably reject all candidates that exceed some threshold. From information theory, Entropy can be interpreted as the average level of *information* or *surprise*, represented by a given distribution. The idea of information entropy

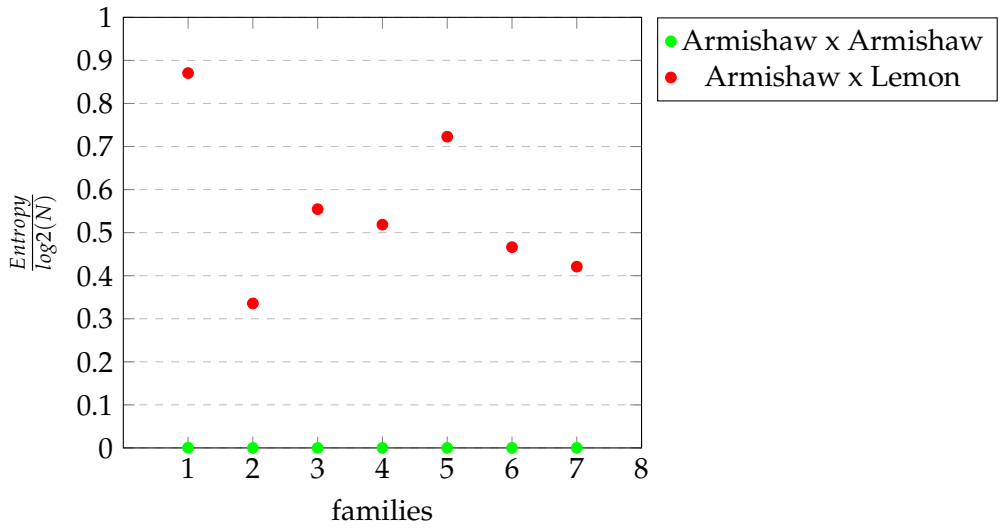


Figure 5.2: **The comparison of ENTROPY values for the `Armishaw` family.** The green data points represent positive linkages and the red represent negative ones. The red comparisons are against **b** sets from the `Lemon family`. This is the resulting entropy values when tested with noise = 0.

was introduced in "A Mathematical Theory of Communication" by Claude Shannon in his 1948 paper [63]. Entropy is defined as

$$S(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i).$$

where x is the random variable, with possible outcomes x_i, each with probability $P(x_i)$, the entropy $S(x)$ of X. We also need to be able to relate thresholds for different choices of n, so divide the entropy by its maximum possible value, $log_2(n)$ so it varies in the range 0 to 1.

5.3.1 How to reject using an ENTROPY threshold

We take the entropy S of the distribution of $P(\mathbf{m}|\mathbf{b})\forall\mathbf{b} \in \mathcal{B}$ (where $\mathcal{B}$ is all of the BDM groups for this surname). We set a rejection threshold T, and where the entropy divided by $log_2(length(\mathcal{B}))$ is greater than T will be rejected. We use the log of the family length so that when we can set an appropriate threshold value that is focused on the uncertainty of the results for each family that is not dependant on the length of the family. Otherwise there would need to be a different T for each different length of $\mathcal{B}$.

5.3.2 An example

In 5.2 the entropy of two last name family comparisons are displayed. The blue is the failure case and the red is the successful one. In the successful case each `Armishaw - Armishaw` comparison is compared to the entropys of when each `Armishaw` family is compared to the Lemon family.

This suggests that we can use entropy to reject negative linkages. By calculating the entropy for a *perfect* case and using it to compare *imperfect* cases like the real `lemons` family, we can rule out cases where the $P(\mathbf{m}|\mathbf{b})$ appears to be significant but is actually not.

There is a very clear division between the entropy of a comparison containing a real match and the entropy of a comparison to an arbitrary family in Figure 5.2.

5.3.3 The flaws of ENTROPY

Entropy assesses *relative* probabilities in $p(\mathbf{b}|\mathbf{m})$ but is blind to *absolute* values in the likelihood $p(\mathbf{m}|\mathbf{b})$, so there is no corollary with the intuitively reasonable (if naive on its own) requirement that *at least one* good match between people should be present for a match between families to be plausible. Even if none of the available **b** are correct we might still get a low entropy due to chance. For example, considering a particular **m**, if one **b** includes a single name that matches while the others do not, the entropy could be low despite all these $p(\mathbf{m}|\mathbf{b})$ being low.

5.4 Better than nothing - BTN

Consider the generative probability $P(\mathbf{b}|m)$ where $P(x|y)$ is the usual Ngrams model based on data in y and m is (just) one of the names in **m** itself. The max $R_{\mathbf{m}} = \max_{m \in \mathbf{m}} \left[P(\mathbf{m} \mid m) \right]$ provides a baseline measure of *how hard a string like* **m** *is to predict, from pieces of itself*. This is a similar concept to the *better than one* method described above.

We consider the ratio between $P(\mathbf{m} \mid \mathbf{b})$ and $R_{\mathbf{m}}$: a value > 1 indicates that **b** provides about as much help in predicting **m** as would one name alone taken from **m** itself. Taking logs yeilds a score

$$S_{\mathbf{m}|\mathbf{b}} = \log P(\mathbf{m} \mid \mathbf{b}) - \log R_{\mathbf{m}}$$

with scores above zero indicating 'suspicious coincidences' in the above sense.

This method compares the probability of a comparison where beta = 0 (so effectively $P(\mathbf{m} \mid \mathbf{null})$) and a "regular" beta = 0.5. The point is that when the $P(\mathbf{m} \mid \mathbf{b})$ is more than the $P(\mathbf{m} \mid \mathbf{null})$ then the B family probably isnt very good. We want a real match to "win" this comparison and it should, because a distribution derived from the real B will almost certainly perform better than using the base distribution.

Chapter 6

Results of the Group-to-Group linkage methods

This chapter assess the accuracy and performance of the group to group linkage methods described in Chapter 4 and the rejection methods described in Chapter 5 on test data. This chapter aims to test each element and to comment on the successes and deficiencies of each method in order to find a suitible combination of linkage and rejection methods. Subject to the noise model described in Section 3.2.2, we would like to answer the following questions:

- Which linkage method is the best (MATCH, NGRAMS)?
- Which rejection method is best (AL1, ENTROPY, BTN)?
- Which value of the ENTROPY threshold is the best?
- How much noise can the algorithm handle?

This chapter is broken up into different sections based on the three different rejection methods. Each rejection method was tested with each of the linkage methods. The same test data set has been used as in the previous chapter.

6.1 The At-Least-1 rejection method - AL1

The AL1 rejection method will reject the linkage when the score from the comparison is worse than when the *m* family is compared to a single one of its members. The AL1 rejection method was applied to both MATCH and NGRAMS linkage methods. The results of both of these linkage methods on test data can be seen in Table 6.1. The results of both of these linkage methods on test data with noise introduced can be seen in Table 6.1. The noisy results are a much more useful predictor of performance on real MLO data than the clean data because we know that the real data is messy. In a noisy situation AL1 was able to reject more negative families than it was in the noiseless case. The best combination was with the NGRAMS linkage method where the precision, accuracy scores are the best. It is still worth noting that it is not performing very well as it is only able to reject 84% of non match families.

6.2 The ENTROPY rejection method

Figure 6.1 the ROC curve (Receiver Operating Characteristic) which plots the rate of True Positive classifications to False Positive classifications is shown at different levels of noise in

linkage	True +	False -	True -	False +	Precision	Accuracy	F1
MATCH	0.996	0.004	0.124	0.876	0.532	0.560	0.694
NGRAMS	1.000	0.000	0.050	0.950	0.513	0.525	0.678

Table 6.1: **Performance of AL1 rejection method with no noise on test data**. Across both of the linkage methods the AL1 method is not able to reject the majority of the true negatives. All of the methods produce low precision and accuracy. The F1 score is higher than the precision and accuracy scores because it is primarily focused on the ability to predict true positives.

linkage	True +	False -	True -	False +	Precision	Accuracy	F1
MATCH	0.740	0.260	0.210	0.790	0.484	0.475	0.585
NGRAMS	0.936	0.064	0.159	0.841	0.527	0.547	0.674

Table 6.2: **Performance of AL1 rejection method on test data with noise = 0.2**. Like on the clean data, the AL1 method is not able to reject the majority of the true negatives when noise is introduced. All of the methods produce low precision and accuracy, lower now that noise has been introduced. Again the F1 score is higher than the precision and accuracy scores because it is primarily focused on the ability to predict true positives.

the test ground truth data set. We use this to understand how the ENTROPY rejection method performs on a range of noisy data. The perfect result would be a curve that has a coordinate of (0,1) and hits the very top left corner as this would mean that it had a 100% True positive and a 0% false positive rate. The NGRAMS model performed very well, and was robust to increased noise in the data. As expected the MATCH linkage method was not robust to noise and whilst performing well on clean data performed poorly on noisy data.

Figures 6.2 and 6.4 show the different entropy values produced for each **m** family when compared to the **b** families with the same surname. Figures 6.3, 6.5 show the same thing but when the experiments are run on a noisy data. Each green point represents a family that does originate in $\mathbf{b} \in \mathcal{B}_{\mathbf{m}}$ (ie. a case we would like to pass) while each red is the same for a different (and wrong) set of families, which we would like to exclude. The families appear on the graph sorted by the positive value, hence the green values decrease from left to right. In setting a threshold for entropy we would want most cases that have high entropy should be excluded (red) while most that are low should ideally pass (green).

6.2.1 The MATCH linkage method with ENTROPY rejection

When the ENTROPY threshold rejection method is used with the MATCH linkage method, we see promising results. ENTROPY produces much better results than the *match* and *al1* rejection methods. On the noiseless test data set Figure 6.2 shows a division between positive and negative families. The green points are *generally* in the lower half of the graph, and the red points are *consistently* in the top half. The entropy values of the MATCH linkage method in Figure 6.2, is linearly separable between matching and unmatched families due to the binary nature of this linkage method. Each comparison is concerned only with the number of identical matches. This means that to score highly there must be many identical matches, and in a comparison with no identical matches there will be a score of zero. In Figure 6.2 we see this the majority of the green points where a real match exists are at the bottom, and the majority of the red points where there is no real match, are at the top. This means that entropy is a suitable measure of family fit. This is good because it means that the data lends itself well to a threshold. It is fairly obvious that an appropriate threshold exists where most

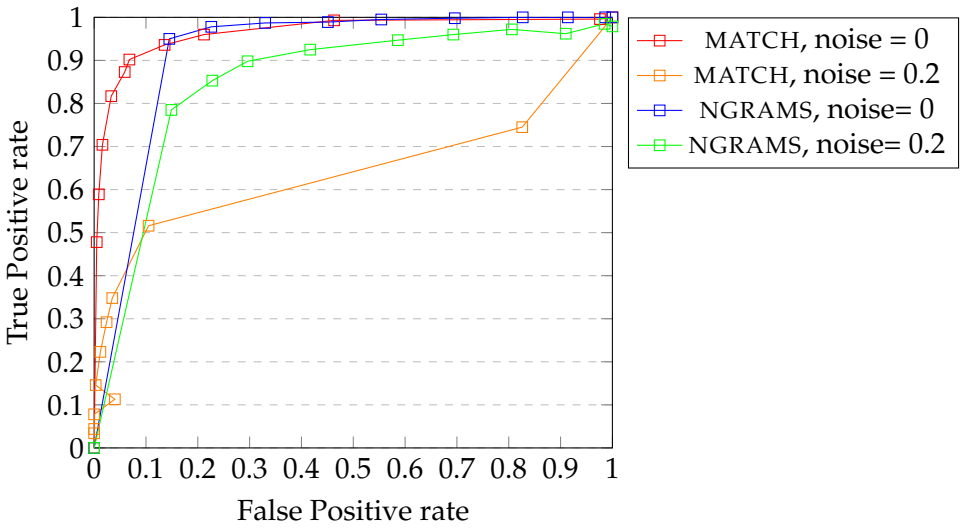


Figure 6.1: **The Receiver Operating Characteristic curves for different linkage methods with Entropy rejection.** The MATCH linkage method with no noise has an almost perfect curve, this can be seen in the red squares. When noise is introduced, performance drops significantly. This is shown in the orange squares. This highlights the fragility of the identical string matches. The best performing linkage method was the NGRAMS model shown in blue. With no noise, the ROC curve is almost as good as the MATCH method with no noise. The model with noise introduced is shown in green. This method performs almost as well as the MATCH and NGRAMS methods without noise.

of the green squares will be accepted and most of the red ones will be rejected. When the experiment is run on noisy data, the ability of the method to reject incorrect comparisons becomes much worse. The noise in the data means that there are a lot less *exact* matches and so the scores are lower. The accuracy values decrease and we can see when comparing Tables 6.3 and 6.4, that this occurs at all threshold values. Interestingly the precision values are higher in the noisy table which means that in general the data points are closer together. This is visible in Figure 6.3, where the red and green points are positioned close together at the top of the graph. Both Figure 6.2 and Figure 6.3 suggest that ENTROPY may be used to distinguish between positive and negative cases.

6.2.2 The NGRAMS linkage method with ENTROPY rejection

When the ENTROPY threshold rejection method is used with the NGRAMS linkage method, we see even more promising results. ENTROPY thresholding produces the best results so far. On the noiseless test dataset Figure 6.4 shows almost most of the positive families at a near zero entropy value and the negative families with much more vertical spread. The green points are *always* at the very bottom of the graph, and the red points are spread out across the whole graph. The real matches in general all have lower entropy values than their corresponding non-match. Like the MATCH linkage method the positive and negative data points are not linearly separable. This suggests that ENTROPY is an appropriate method for rejection. We can eliminate a significant proportion of non matches while still accepting most of the real matches. Tables 6.3 and 6.4 show that an appropriate threshold exists where most of the green points will be accepted and some of the red points will be rejected. On noisy data, the ability of the method to reject incorrect comparisons decreases. The accuracy values decrease and we can see when comparing Tables 6.5 and 6.6, that this occurs at all

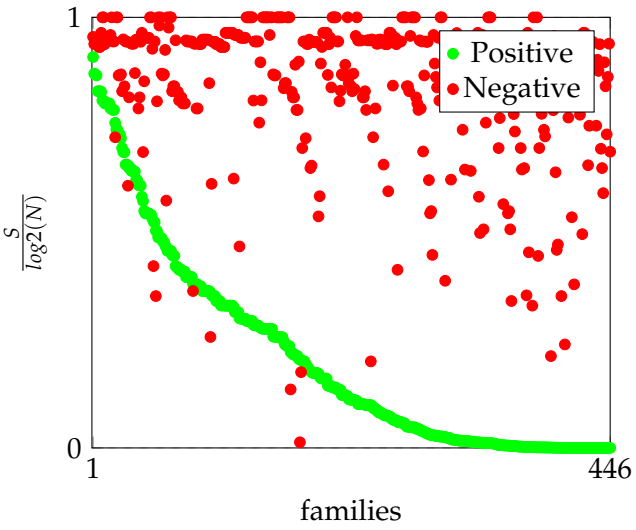


Figure 6.2: **Comparison of the entropy for each family, using the MATCH linkage method, noise = 0.** The data points (one per family) are sorted by the green value. The values are plotted using a scale of $\frac{S}{log2(N)}$ so that they would all be in the range of zero to one. The positive (green) points are generally in the lower half of the graph, and the red points are consistently in the top half. Note the lack of correlation with the red values.

threshold values. The best accuracy is 0.9 with a threshold value of 0.1 when noise is not present in the data. This only decreases to 0.82 with the same threshold value when noise is introduced. Noise is at a level of 0.2 as per Section 3.2.2 Both Figure 6.4 and Figure 6.5 suggest that ENTROPY may be used to distinguish between the positive and negative cases.

6.2.3 What effect did the noise have?

Figures 6.3 and 6.5 show the different values of entropy produced for each **m** family when compared to the **b** families with the same surname when noise is introduced. Adding noise to the test data caused some of the entropy values to increase - this occurred for all of the linkage methods. It is particularly obvious for the MATCH linkage method where in Figure 6.2 there is a fairly even distribution of families top and bottom, whereas in Figure 6.3 when is noise present there is a high concentration of families at the top of the graph and very few at the bottom. This means that there is much more uncertainty in the distribution of surname group. Now, instead of having low entropy the red squares (the real comparisons) are more uncertain and have drifted to the top. This is likely to be because the noise introduces spelling mistakes and removes names completely and identical matches that have been removed. This means that the clear advantage that the true match would have had has been reduced.

With the noisy NGRAMS model, the behaviour is similar to the noiseless MATCH model.The accuracy and precision scores shown in Table 6.6 are still pretty high.

6.3 The Better Than Nothing Method - BTN

The BTN rejection method, described in Section 5.4, compares each family-family comparison to what would have happened if the same had been generated by the base distribution. A result will be rejected when the score from the comparison $P(\mathbf{m}|\mathbf{b})$ is worse than the com-

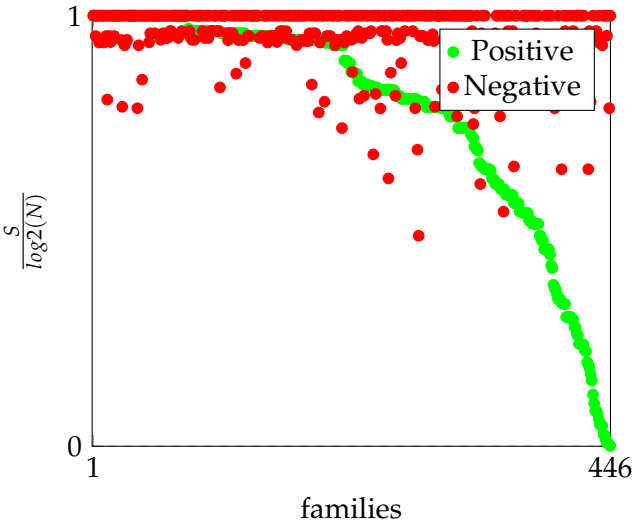


Figure 6.3: **Comparison of the entropy for each family, using the MATCH linkage method, noise = 0.2.** The positive green points are generally in the lower half of the graph, and the red points are consistently in the top half. However by introducing noise about half of the green points are now amongst the red. This will be because noise will mess up name strings and so there will no longer be identical matches.

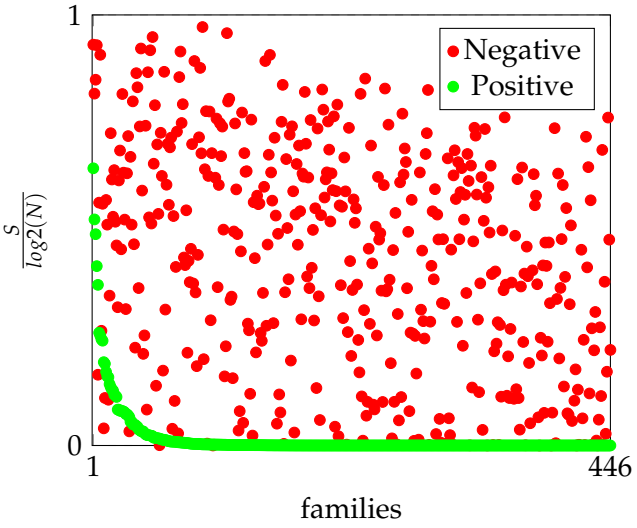


Figure 6.4: **Comparison of the entropy for each family, using the NGRAMS linkage method, noise = 0.** Almost all of the positive green points sit at the very bottom of the graph, and the red points are spread out fairly evenly from 0 to 1.

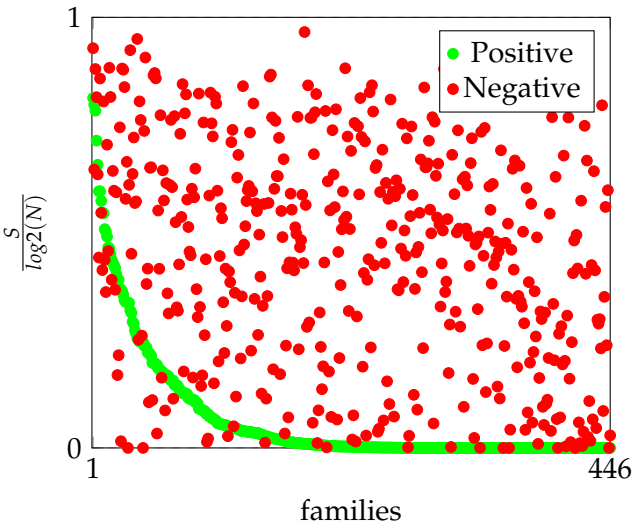


Figure 6.5: **Comparison of the entropy for each family, using the NGRAMS linkage method, noise = 0.2.** Almost all of the positive green points sit at the very bottom of the graph, and the red points are spread out fairly evenly from 0 to 1. Compared to the noiseless model, there are more high entropy positive green points, this can be seen in a longer slope at the start of the green curve.

parison of the same **m** family drawn only from the base distribution, rather than from the **b** family i.e. $P(\mathbf{m}|null)$.

This rejection technique does not apply to the MATCH linkage method. The MATCH linkage method looks only at exact matches and does not draw from the base distribution or use Ngrams at all. We have applied the BTN model to the NGRAMS linkage model. Figure 6.6 shows the distribution of scores for both the positive and negative linkages on our test ground truth data set. This shows that BTN threshold is able to distinguish between the two cases most of the time.

When this method was first presented in Section 5.4 we proposed that any comparison "with scores above zero indicating 'suspicious coincidences' in the above sense." should be rejected. However it appears that there might be some better results if the threshold were set higher. This suggests that the ratio between $P(\mathbf{m} \mid \mathbf{b})$ and $R_{\mathbf{m}}$ might be somewhat larger than 1.

The BTN rejection in combination with the NGRAMS linkage method performed very well (see Tables 6.7 and 6.8). The NGRAMS linkage method produced a best accuracy value of 0.957 (threshold = 0.1) and a precision value of 0.921 at the same threshold on non noisy data. On noisy data the NGRAMS linkage method produced a best accuracy value of 0.851 (threshold = 0.1) and a precision value of 0.914 at the same point. The BTN method appears to correctly reject almost all of the non matches and does not interfere with the true cases. These Tables also show that the introduction of noise is not creating significant disruption in the scores, as the accuracy is only decreasing by less than 0.1 when this very significant amount of noise is introduced.

The BTN rejection method produces much smoother results than ENTROPY. This method is by far the best performing rejection method analysed in this chapter. The ROC curve shown in Figure 6.7 lets us compare the performance of the methods. This visualises what Tables

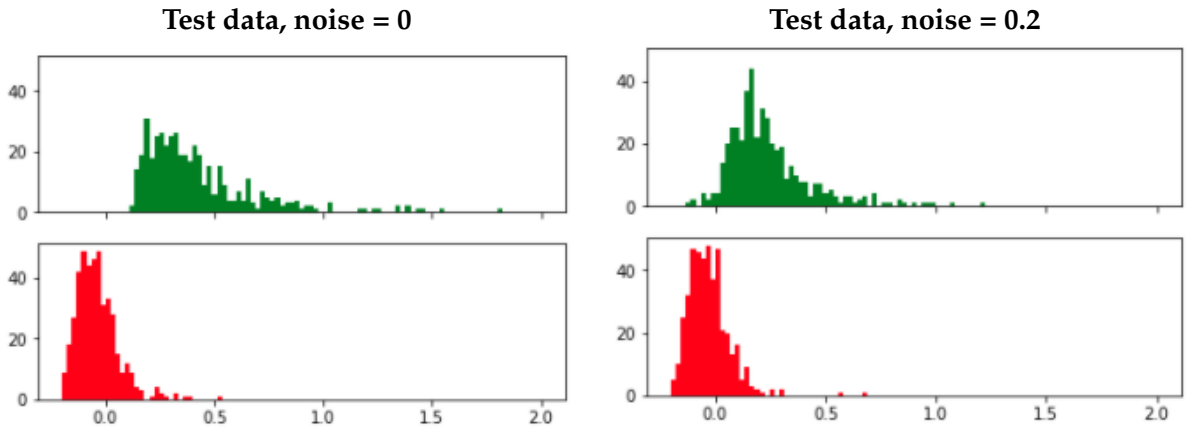


Figure 6.6: **Histograms of scores from NGRAMS linkage method with BTN rejection.** Histograms of scores using the BTN method 5.4. The BTN rejection method in the no noise situation *(left)*, and the same approach with noise = 0.2 *(right)*. Both have n = 4 and $\beta = 0.5$. The green, like in previous examples are comparisons where the real family is present and the red are rejection comparisons where the real family has been replaced by a randomly generated one. There is a clear difference between positive and negative cases for both noisy and clean cases.

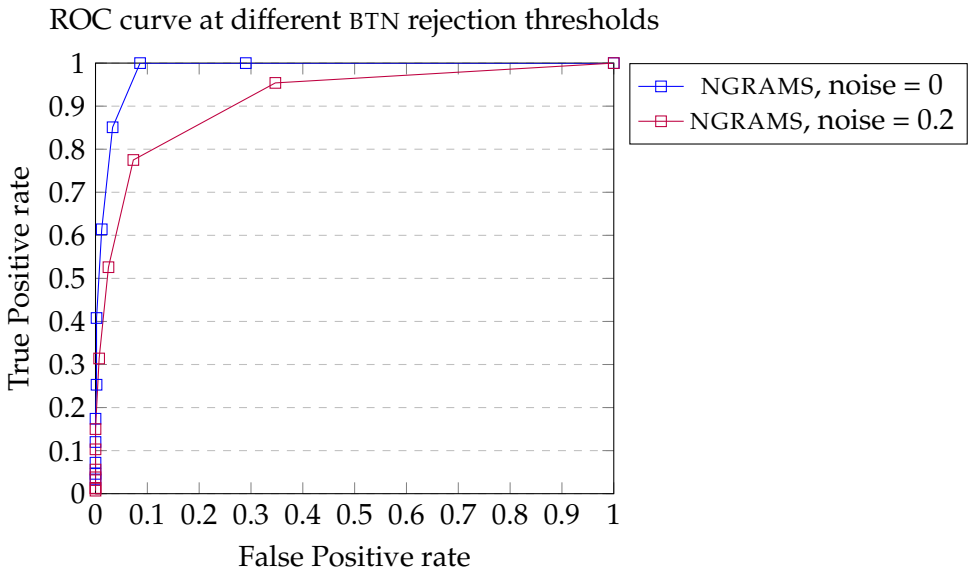


Figure 6.7: **The Receiver Operating Characteristic curve for NGRAMS linkage methods with BTN rejection.** All of the the ROC curves in the diagram are very promising. The NGRAMS linkage method performs well, Performance is good both with clean data and when noise is introduced. This is shown in the blue and the purple lines.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.1	0.478	0.522	0.995	0.005	0.989	0.736	0.644
0.2	0.589	0.411	0.991	0.009	0.984	0.790	0.737
0.3	0.704	0.296	0.984	0.016	0.978	0.844	0.819
0.4	0.817	0.183	0.967	0.033	0.961	0.892	0.883
0.5	0.873	0.127	0.941	0.059	0.937	**0.907**	0.904
0.6	0.902	0.098	0.914	0.086	0.913	0.908	0.907
0.7	0.936	0.064	0.864	0.136	0.874	0.900	0.904
0.8	0.960	0.040	0.788	0.212	0.819	0.874	0.884
0.9	0.993	0.007	0.537	0.463	0.682	0.765	0.809
1.0	0.996	0.004	0.024	0.976	0.505	0.510	0.670

Table 6.3: **Performance of ENTROPY rejection method on test data, using the MATCH linkage method with no noise.** The threshold uses a value of $\frac{S}{log2(N)}$ so that all of the values are in the range zero to one. The best accuracy occurred at a threshold value of 0.6, however there was also high accuracy at 0.4, 0.5 and 0.7. Generally this method was able to correctly reject negative cases.

6.4 Summary

The best combination with the linkage method NGRAMS was the with ENTROPY rejection method. The NGRAMS-AL1 combination was not able to correctly reject the false families, and produced a very small number of True Negative results. The ENTROPY results are very different between the different methods. The threshold for the MATCH linkage method for the best precision/accuracy with noisy data is 0.9, whereas the threshold is a lot lower for NGRAMS linkage performing best at 0.1.

The MATCH linkage method is very good at identifying True Positives but very bad at rejecting True Negatives or handling noise when it was added with all of the rejection methods. The MATCH linkage method was only applicable to the AL1 and the entropy rejection methods.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.1	0.034	0.966	1.000	0.000	1.000	0.517	0.066
0.2	0.044	0.956	1.000	0.000	1.000	0.522	0.084
0.3	0.078	0.922	1.000	0.000	1.000	0.539	0.145
0.4	0.113	0.887	0.996	0.004	0.966	0.554	0.202
0.5	0.146	0.854	0.997	0.003	0.981	0.572	0.255
0.6	0.223	0.777	0.988	0.012	0.949	0.606	0.361
0.7	0.292	0.708	0.976	0.024	0.924	0.634	0.444
0.8	0.348	0.652	0.965	0.035	0.909	0.657	0.504
0.9	0.516	0.484	0.895	0.105	0.831	**0.705**	0.636
1.0	0.745	0.255	0.174	0.826	0.474	0.460	0.580

Table 6.4: **Performance of ENTROPY rejection method on test data, using the MATCH linkage method with noise = 0.2.** The best accuracy occurred at a threshold value of 0.9. The noise in the test data significantly impacted the MATCH methods ability to correctly reject negative cases. The accuracy numbers are much lower at every threshold value.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.1	0.950	0.050	0.855	0.145	**0.868**	**0.902**	0.907
0.2	0.978	0.022	0.774	0.226	0.812	0.876	0.887
0.3	0.987	0.013	0.670	0.330	0.749	0.828	0.852
0.4	0.989	0.011	0.549	0.451	0.687	0.769	0.811
0.5	0.995	0.005	0.446	0.554	0.642	0.721	0.781
0.6	0.998	0.002	0.304	0.696	0.589	0.651	0.741
0.7	1.000	0.000	0.173	0.827	0.547	0.586	0.707
0.8	1.000	0.000	0.086	0.914	0.522	0.543	0.686
0.9	1.000	0.000	0.014	0.986	0.504	0.507	0.670
1.0	1.000	0.000	0.000	1.000	0.500	0.500	0.667

Table 6.5: **Performance of ENTROPY rejection method on test data, using the NGRAMS linkage method with no noise.** The best accuracy (0.902) occurred at a threshold value of 0.1. Performance decreased as the threshold increased. The NGRAMS method was able to reject negative linkages.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.1	0.785	0.215	0.851	0.149	0.840	**0.818**	0.812
0.2	0.853	0.147	0.772	0.228	0.789	0.812	0.820
0.3	0.898	0.102	0.704	0.296	0.752	0.801	0.818
0.4	0.925	0.075	0.583	0.417	0.690	0.754	0.790
0.5	0.947	0.053	0.414	0.586	0.618	0.680	0.748
0.6	0.960	0.040	0.307	0.693	0.581	0.634	0.724
0.7	0.972	0.028	0.194	0.806	0.547	0.583	0.700
0.8	0.962	0.038	0.090	0.910	0.514	0.526	0.670
0.9	0.985	0.015	0.015	0.985	0.500	0.500	0.663
1.0	0.979	0.021	0.000	1.000	0.495	0.490	0.657

Table 6.6: **Performance of ENTROPY rejection method on test data, using the NGRAMS linkage method with noise = 0.2.** The best accuracy (0.818) occurred at a threshold value of 0.1. Performance decreased as the threshold increases. The NGRAMS method was able to reject negative linkages. The noise in the test data did not significantly impact the MATCH methods ability to correctly reject negative cases. The accuracy numbers are only slightly lower at each threshold value.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	1.000	0.000	0.710	0.290	0.775	0.855	0.873
0.1	1.000	0.000	0.914	0.086	**0.921**	**0.957**	0.959
0.2	0.851	0.149	0.967	0.033	0.963	0.909	0.904
0.3	0.614	0.386	0.988	0.012	0.980	0.801	0.755
0.4	0.408	0.592	0.998	0.002	0.994	0.703	0.578
0.5	0.253	0.747	0.998	0.002	0.991	0.625	0.403
0.6	0.174	0.826	1.000	0.000	1.000	0.587	0.296
0.7	0.120	0.880	1.000	0.000	1.000	0.560	0.215
0.8	0.072	0.928	1.000	0.000	1.000	0.536	0.134
0.9	0.046	0.954	1.000	0.000	1.000	0.523	0.088
1.0	0.033	0.967	1.000	0.000	1.000	0.517	0.065

Table 6.7: **Performance of BTN rejection method on test data, using the NGRAMS linkage method with noise = 0.** As the threshold value increases the precision decreases. Aside from when the threshold is at zero, the accuracy decreases as the threshold increases. The best Accuracy is at a threshold value of 0.1.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.954	0.046	0.653	0.347	0.733	0.803	0.829
0.1	0.775	0.225	0.927	0.073	0.914	**0.851**	0.839
0.2	0.526	0.474	0.975	0.025	0.954	0.750	0.678
0.3	0.314	0.686	0.993	0.007	0.977	0.653	0.475
0.4	0.150	0.850	1.000	0.000	1.000	0.575	0.261
0.5	0.103	0.897	1.000	0.000	1.000	0.552	0.187
0.6	0.056	0.944	1.000	0.000	1.000	0.528	0.107
0.7	0.038	0.962	1.000	0.000	1.000	0.519	0.074
0.8	0.012	0.988	1.000	0.000	1.000	0.506	0.023
0.9	0.013	0.987	1.000	0.000	1.000	0.506	0.026
1.0	0.007	0.993	1.000	0.000	1.000	0.504	0.015

Table 6.8: **Performance of BTN rejection method on test data, using the NGRAMS linkage method with noise = 0.2.** The performance is much the same when noise is introduced. The best Accuracy is 0.851 when the threshold is 0.1, shown in the bold print.

Chapter 7

Methods for finding the best alignment

This chapter motivates and outlines a way forward in addressing the full alignment problem. We will be linking groups in MLO to ones in BDM by looking at the best alignments between the groups. In practice it is also, of course, very natural to think of a specific alignment when considering the plausibility of one group versus another: "What if `James` in **b** is really `Jimmy` in **m**?" and so on. The main advantage of a solution built in terms of specific alignments is that it ensures that an individual is only accounted for once: "If the `James W` in **b** also had the middles names of `Andrew` and `David`, he could in theory be counted 3 times when compared with a family in **m** with 3 brothers: `James`, `Andrew` and `David` ".

If the data set is trustworthy and it is believed that is has all the right people just misspelled then looking for the best alignment is both possible and of advantage. However if it is believed that (one or both) of the data sets are of poor quality then choosing a specific alignment could decrease the ability to find the correct family.

7.1 With alignments

What is the probability of some set of names **m**, given they originate from a set of (named, identified) people **b**? We are interested in the actual alignment of individuals because we want to know if each person in **m** is in **b**.

By the sum rule of probability,

$$P(\mathbf{m} \mid \mathbf{b}) = \sum_{\mathbf{z} \in \mathcal{Z}} P(\mathbf{m}, \mathbf{z} \mid \mathbf{b}) \tag{7.1}$$

$$= \sum_{\mathbf{z} \in \mathcal{Z}} P(\mathbf{m} \mid \mathbf{z}, \mathbf{b})\ P(\mathbf{z} \mid \mathbf{b}) \tag{7.2}$$

Note we can drop the **b** in the second reduces to just B, leaving

$$\underbrace{P(\mathbf{m} \mid \mathbf{b})}_{F} = \sum_{\mathbf{z} \in \mathcal{Z}} \underbrace{P(\mathbf{m} \mid \mathbf{z}, \mathbf{b})}_{f(z)}\ \underbrace{P(\mathbf{z} \mid B)}_{p(z)} \tag{7.3}$$

These 3 probabilities are worth giving shorter "names" to, for future reference.

- $F = P(\mathbf{m} \mid \mathbf{b})$ the *marginal likelihood* is what we want to find.
- $f(z) = P(\mathbf{m} \mid \mathbf{z}, \mathbf{b})$ is the *likelihood* of the names, given source **b** and alignment **z**.

- $p(z) = P(\mathbf{z} \mid \mathbf{b})$ is the prior over alignments.

The quantity we need to calculate is therefore

$$F = \sum_{\mathbf{z} \in \mathcal{Z}} f(z)\ p(z) \tag{7.4}$$

Because this is represents what PKW are trying to do when they are looking for missing shareholders. If we know F (up to a proportionality constant) for each $\mathbf{b} \in \mathcal{B}$ we can readily [1] find the quantity we're really interested in: $P(\mathbf{b}|\mathbf{m})$.

Even without specifying $f(z)$ and $p(z)$ in detail though, we can look at what is required for inference, namely the computation of F.

7.1.1 Why is this hard?

In reality the whakapapa or family structure and the gender of a baby determines what name it gets. PKW need to identify the real identities of shareholders who have money to collect. We need to be able to infer these identities at scale (the whole of MLO) but only. If we can model this scenario/world/situation then we can get the posterior distribution (the probability of the names given the sets). At an individual level this could be perceived as simple however, the actual inference problem is massive and non-trivial. It is the type of task that is difficult even for the human expert: Their choices rely on a variety of separate data sets, years of experience and an knowledge of Māori whakapapa.

We can calculate the probability of two sibling groups given the names inside. By using Bayes inference, we can invert this probability with Bayes theorem and get the probability of alignment given the two sets. Calculating $f(z)$ is hard. This is because we do not know which person from each group is from the other group. This means that naively there are at least $\binom{n}{m}$ combinations plus additional alignments where people do not match at all and match to a person outside of the group. This is a huge number of sets to check and does not scale at all.

The brute force approach is to find F exactly, by calculating the whole sum. This means working out all of the different possible alignments/linkages and then calculating the $p(x)$ for each alignment.

The main problem with this is that there are a lot of possible linkages $\mathbf{z} \in \mathcal{Z}$: When $|\mathbf{b}| = |\mathbf{m}|$ there are $|\mathcal{Z}| = |\mathbf{b}|!$ because there are $|\mathbf{b}|!$ different permutations of $\mathbf{m}$ to try against $\mathbf{b}$. When $|\mathbf{b}| \neq |\mathbf{m}|$ something like $|\mathcal{Z}| =^{\mathbf{b}} C_{\mathbf{m}}$.

The brute force approach will only be feasible when either B or M is small and so the resultant sum is also small. We have included the brute force approach in explanation only and its results are not included in this thesis because they are always correct, not very interesting and incredibly slow.

7.2 Model for $f(z)$

Our generative model, $f(z)$ is the likelihood of a set of names, given source $\mathbf{b}$ and alignment $\mathbf{z}$. In order to build $p(\mathbf{m}|\mathbf{b}, \mathbf{z})$, and from there to $p(\mathbf{m}|\mathbf{b})$, which is $\propto p(\mathbf{b}|\mathbf{m})$, we need models for $p(m|b)$ and $p(m|\texttt{null})$. $p(m|\texttt{null})$ lets us represent a linkage structure in which given an individual in MLO there is no relevant record in BDM.

[1] ie. by normalising over all the F values arrived at for $\mathbf{b} \in \mathcal{B}$

If names in MLO, *given birth name*, are independent of the names of siblings, then this factorizes as follows:

$$f(z) = P(\mathbf{m} \mid \mathbf{z}, \mathbf{b}) \tag{7.5}$$

$$= \prod_{j=1}^{M} P(m_j \mid b_{z_j}) \tag{7.6}$$

Working in log space has numerical advantages, because we can add instead of multiply. It is more convenient to work with $S = \log f$ and use this as a *score* instead,

$$S(\mathbf{z}) = \sum_{j=1}^{M} S_{j,z_j} \tag{7.7}$$

where

$$S_{j,z_j} = \log P(m_j \mid b_{z_j}) \tag{7.8}$$

$|\mathcal{Z}|$ is the total number of all possible alignments. Unfortunately $|\mathcal{Z}|$ is really large and not feasible to compute at the scale of the approximately 5 million records in MLO and BDM. In this section we propose and compare several methods for calculating $|\mathcal{Z}|$ more efficiently.

The idea is to use the f of the best alignment we can find as a proxy for F.

$$F \approx \quad f(\mathbf{z}^\star) \tag{7.9}$$

where $\mathbf{z}^\star$ is the *optimal* alignment:

$$\mathbf{z}^\star = \arg\max f(\{\mathbf{z}\}) \tag{7.10}$$

recalling that the f in question is $f(\mathbf{z}) = \Pr(\mathbf{m}|\mathbf{b}, \mathbf{z})$.

Note that since log is monotonic, $\arg\max_{\mathbf{z}} f(\mathbf{z}) = \arg\max_{\mathbf{z}} S(\mathbf{z})$ which means finding $\mathbf{z}^\star$ has a completely "additive" character since $S(\mathbf{z}) = \sum_i S_{i,z_i}$.

7.2.1 Optimising $f(\mathbf{z})$

Instead of using costly compute and trying every single combination of alignments we can approximate this using *Dynamic Programming*. Dynamic programming is a form of Bellman's equation which is essentially the same as the *Viterbi* algorithm [64]. The *Viterbi* algorithm finds "the most likely sequence of states of the hidden chain X which might have given rise to a given set of observations." [66].

There are already many dynamic programming algorithms that have been developed to do sequence alignment - a similar problem often seen in genetic sequencing and other string matching or word problems [67]. Many of these algorithms add spaces to the string so that letters match up. For example consider if we wanted to match the two strings `ABCDDE` `ABBDE`. We would get:

```
AB_CDDE

ABB__D_E
```

Although at first glance this technique might be applicable to our scenario it relies on having a good ordering for individuals so that the 'gaps' could be inserted and individuals paired up like the letters were in the example above. In theory, arranging the individuals

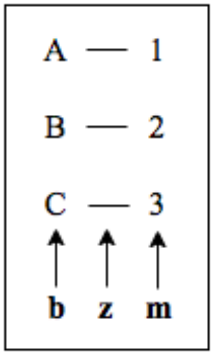


Figure 7.1: **Example alignment.** Here `[A,B,C]` is the **b** family, `[1,2,3]` is the **m** family and **z** is a possible alignment between them.

alphabetically would work except for the fact that this would then be very brittle to nicknames as they would move the individual out of sequence and never be aligned. We have instead developed our own dynamic programming algorithm.

Dynamic programming will provide us with a method for finding the best alignment, with less computational pain. We will be able to find the best global alignment between the two groups (MLO and BDM). This is possible because the alignment problem we face has optimal substructure. "A problem is said to have optimal substructure if an optimal solution can be constructed from optimal solutions of its sub problems" [54].

7.2.1.1 Optimal substructure

Suppose that we have a set of BDM names **b** =[`A,B,C`] and a set of MLO names **m** = `[1,2,3]`. So there are 9 possible links: {`A1,A2,A3,B1, B2, B3, C1,C2,C3`} The obvious linkage structures are not too numerous, being just these six:

$$[\{\texttt{A1,B2,C3}\},\{\texttt{A1,B3,C2}\},\{\texttt{A2,B1,C3}\},\{\texttt{A2,B3,C1}\},\{\texttt{A3,B1,C2}\},\{\texttt{A3,B2,C1}\}]$$

However there is also the case where an element in **m** does not match any of the elements in **b**. For each individual we also have to have an null person, in case there is none in **b** that is also in **m**.This means that there are really 6x6 options instead of 3x3. When the sizes of the sets are small there are a low number of linkage structures to consider. In our small example of MLO length of 3, there are only 36 different linkage structures. However for larger set sizes this number is much larger and calculating them would become tedious.

Suppose the link structure $L = \{\texttt{A1,B2,C3}\}$, is actually the optimal alignment. We can be sure that we will get the optimal alignment dynamically by making use of the core principal of dynamic programming - Optimal substructure.

The match between `C` and `3` is in the L set. Consider the sub problem before `C3` was added. It must have been a linkage involving the subset of links {`A1, A2, B1, B2`}. Now, since L is optimal, whatever the linkage in that sub-list, it must also be optimal, for that subset i.e. it's the best linkage involving the terminals `A, B` for BDM and `1, 2` for MLO, otherwise L couldn't have been optimal in the first place. Thus an example of the optimal substructure required for dynamic programming to be a good fit.

So using this concept of optimal substructure this algorithm allows us to find the best alignment between two sets of names/people: Suppose we count additions to the set size via a counter t.

For each linkage (e.g. A1), we assign a value: R(A1), which is a valuation or score of that link.

We want the total valuation of a linkage structure to be:

$$P(\mathbf{m} \mid \mathbf{b}, \mathbf{z}) = \prod_{j=1}^{M} p(m_j \mid b_{z_j})$$

Working in log space, the (log) valuation is therefore:

$$\sum_{j=1}^{M} \log p(m_j \mid b_{z_j})$$

And so the "local" valuation of the link, must be $\log p(m_j \mid b_{z_j})$. For instance we'll have $R(\texttt{A1}) = \log p(m_1 \mid b_A)$.

We are interested in finding the set with the highest possible sum total of R values which will be $logP(\mathbf{m}|\mathbf{z}, \mathbf{b})$. At *t=1* we have the list of possible links {A1, A2, A3, B1, B2 ... C3}, these being all of the different options for the first element in the set. Each of these 9 alignments has a list, consisting of only itself, eg. $list(\texttt{A1}, t = 1) = [\texttt{A1}]$. This will eventually grow as t increases. And each also has a value V, which is the cumulative value of its list. To start with *t=1* all of the lists have just the one element, so $V(\texttt{A1}, t = 1) = R(\texttt{A1})$ and so on. For *t=2*, we want to consider adding a second element to the set of links. We don't know which will be optimal overall but can find the value of any combination formed thus far. For example when $V(\texttt{B2}, t = 2)$, we would consider the V of each of the *t=1* options, and pick the maximum value. However because in an alignment each person is only linked to a maximum of one other person, we must exclude anything that mentions B2 in its list. At *t=2* this is just B2 itself. In this simple example, that just leaves A1, A3, C1, C3. We choose the max "source" node from *t=1*. Suppose it's A1 (not A3, C1 or C3). In that case, we would write $V(\texttt{B2}, t = 2) = V(\texttt{A1}, t = 1) + R(\texttt{B2})$, because this is the value of the linkage set {A1,B2}, $list(\texttt{B2}, t = 2) = list(\texttt{A1}, t = 1)$ with B2 appended to it. This generalises to all subsequent t. At the end, we pick the node with the largest value of V, and its list will be the optimal alignment. This means that the optimal alignment for a group of size N is the same as the optimal alignment for a group of size N-1 plus the Nth item aligned with its pair.

We can find the best single alignment much more quickly than integrating over all of them:

$$F \approx \quad f(Z_{\text{best}}) \tag{7.11}$$

7.2.1.2 The BEST ALIGNMENT algorithm for aligning two sets of names

The BEST ALIGNMENT algorithm works by selecting all of the initial one-to-one combinations of the two groups and then greedily finding the next best combination to add to each such alignment. Continuing until there are no more possible additions in each alignment, we arrive at the best overall.

The BEST ALIGNMENT algorithm uses the following components:

- $\mathbf{m} = [m_1, m_2, \ldots m_M]$ a list of people from MLO, defined earlier in Section 3.2.
- $\mathbf{b} = [b_1, b_2, \ldots b_B]$ a list of people from BDM defined earlier in Section 3.2.
- G = The relative values of matching each item in the first group to each item in the second group.

- L - a list of the different alignments. This is initialised as all of the size 1 combinations. E.g. For the two sets `A,B,C` and `1,2,3` $L = [$`A1, A2, A3, B1, B2, B3, C1, C2, C3`$]$.
- *Mask* - A list of matrices which hold the values which are available to be selected by the algorithm, there is one for each different alignment.
- V - The relative values of matching each item in the first group to each item in the second group.
- W - All of the valid values that are available to be chosen. This is found by multiplying V by the *Mask*.
- t - the count individuals selected as part of the alignment.

The BEST ALIGNMENT algorithm begins by calculating L, by storing multiple different alignments. This creates a list of *Mask*s which hold the valid values for each different alignment. As we iterate through t, we build up the alignments in L by calculating which values V are $validValues$ and choosing the best available option.

Algorithm 1 Dynamic program for finding BEST ALIGNMENT between b and m groups

```
input: MLO group m, BDM group b
Result: Best alignment between m, b
M,N = length(m), length(b)
 mask ← List[List[NxM]]
 L ← List[List[alignmentTuples]]
 for t ← 1 to min(N, M) − 1 do
    mask = updateMask(mask,L)
     V = recalcV(L)
     newV = MxN array of floats
     for r ← 1 to M do
        for c ← 1 to N do
           W = V .* mask[r,c]
            bestValue = maximum(W)
            bestCoord = argmax(W)
            if bestVal > 0 then
               newV[r,c] = bestVal + G[r,c]
                append(L[r,c], (bestCoord))
           else
              L[r,c] = []
           end
        end
    end
    V = newV;
end
return L[indexOf(Max(V))]
```

7.2.2 Proof

We can prove that the algorithm returns the optimal result by induction:

Proof. Because L contains all sets of size 2 from both groups, the optimal solution must be present at this point.

General case: Assuming that L contains the optimal solution at size N, at size $N+1$, the new list L must also contain the most optimal solution because the update step is done greedily for each combination there is no better solution that could exist at size $N+1$. □

7.3 Methods for measuring $P(m_i|b_j)$

So what is a good form, for $p(m_i|b_j)$ where m_i and b_j are names from **m** and **b**? Between b_j and m_i a lot can happen. First note the contexts were very different (one dominated by compliance with the crown's definition of legal identity, the other with connection to whenua). Then there are shortenings, additions (some predictable, others entirely new), plus flawed memories, alternative spellings and plain typos, to name just some of the effects. In coming up with a tractable procedure we will not be able to model each of these possibilities explicitly.

Some nomenclature for alignments:

- *Alignment*. This is defined as birth-origin indices for elements of **m**. $z_j = i$ corresponds to the assertion that the j^{th} element of **m** is actually the same person as the i^{th} element of **b**.
  - We can call the whole vector **z** an **alignment** bewtween sets of **b** and **m**.
  - Note that $|\mathbf{z}| = |\mathbf{m}| = M$.
  - We also need a notation to represent the *lack* of a link from **b**: let $z_j = \texttt{null}$ represent the assertion that the j^{th} element of **m** does not correspond to anybody from **b**.

In this section we will outline two possible methods for this: EDIT DISTANCE and NGRAMS.

7.3.1 Probabilistic adaptation of EDIT DISTANCE

Our model for $f(z)$ is the likelihood of the names, given source **b** and linkage **z**.

$$f(z) = P(\mathbf{m} \mid \mathbf{z}, \mathbf{b})$$

$f(z) = P(\mathbf{m} \mid \mathbf{z}, \mathbf{b})$ is the *likelihood* of the names, given source **b** and linkage **z**

The first term is the probability that the alignment of M generated from B. This uses *Levenshtein distance*, also known as the *edit distance*. Edit distance

In their 1974 original paper proposing an iterative solution to the dynamic problem, Wagner and Fischer describe edit distance as "The string-to-string correction problem is to determine the distance between two strings as measured by the minimum cost sequence of "edit operations" needed to change the one string into the other."[55]

$$lev_{a,b}(i,j) = \begin{cases} max(i,j) & \text{if } \min(i,j) = 0 \\ min \begin{cases} lev_{a,b}(i-1,j) + 1 \\ lev_{a,b}(i,j-1) + 1 \\ lev_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise} \end{cases} \tag{7.12}$$

Levenshtien distance operations include substitutions, insertions and deletions. Edit distance has many applications such as finding the distance or difference between 2 words or in error correction. Dynamic programming is a common method of producing the *edit distance /Levenshtien distance*.

Where for a given alignment there are N name pairs, $P(\mathbf{m}|\mathbf{z},\mathbf{b})$ can be defined as follows:

$$P(\mathbf{m} \mid \mathbf{z},\mathbf{b}) = \prod_{i=0}^{N} P(\mathbf{m}_{\mathbf{z}_i} \mid \mathbf{b}_i) \tag{7.13}$$

$$= \prod_{i=0}^{N} \text{P(namesEqual)} \tag{7.14}$$

$$P(\mathbf{m}_i \mid \mathbf{z},\mathbf{b}_i) = \text{P(namesEqual} \mid \text{ Gender)} \tag{7.15}$$

We need a prior for the way in which names change. Names are compared using a combination of edit distance as well as a prior distribution of names changes in New Zealand. This used a reported value of 7000 names changes to the Department of Internal Affairs per year. This value was for males and females of all ages but did not include people who had married as they are automatically allowed to use their married name without a submission to the DIA [56]. This statistic has been used as a prior for name change probabilities for first names. Let ρ represent a name change. Using the current New Zealand population this becomes:

$$P(\rho) = 7000/4794000 = 0.00146$$

So now we can work out for a name pair what the probability is that the first names and last names equal given the gender of the person. Let N be the string that represents a person's name. In the formula below we are comparing two people: i and j. This is just:

$$P(N_i = N_j) = (1-\rho) * \delta(\mathbf{b}_i, \mathbf{m}_j) \; + \; \rho * lev(\mathbf{b}_i, \mathbf{m}_j) \tag{7.16}$$

where the "Kronecker delta" δ is used to select exact matches:

$$\delta_{i,j} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if otherwise.} \end{cases} \tag{7.17}$$

7.3.2 NGRAMS

We could use the same NGRAMS method as described in Section 4.2. However there are a few minor changes that need to be made to the algorithm. The basic principal of Ngrams is to ask the question "is this word likely to have come from this distribution?". We are still trying to achieve this in the alignments approach as well, except that the distribution of interest is not the whole of **b** but instead a single name. In order to use NGRAMS in the dynamic method we need to be able to apply it to single words in **m**. Instead of looking at the Ngrams distribution of the whole **b** family to predict the **m** name, we use only the Ngrams distribution of the specific **b** name.

7.4 Summary

In this chapter we have introduced a method for finding the optimal alignment between two groups of individuals from BDM and MLO. This is done using dynamic programming and when discussed is called the BEST ALIGNMENT method. We have also discussed two different methods for calculating $P(m_i|b_j)$ which is a measure for how different two names are. The first is EDIT DISTANCE which calculates how many letters would need to be altered to make m_i from b_j. The second is NGRAMS which uses the same principles as described in Section 4.2 to model the probability of generating m_i using an ngrams distribution build from the **b** family.

Chapter 8

Experiments and Results - with best alignment

We want to assess the accuracy of linkages arrived at using a single ”best” alignment. This chapter aims to test each element of this process and comment on the successes and deficiencies of each method. We are testing:

- Which method EDIT DISTANCE, NGRAMS to use for measuring $P(\mathbf{m}|\mathbf{b})$?
- Which method ENTROPY or BTN performs best for rejections?
- How robust are the models to noise?

8.1 Linkage Results without rejections

This section compares the accuracy of the BEST ALIGNMENT found using EDIT DISTANCE versus NGRAMS to measure $P(\mathbf{m}|\mathbf{b})$. It is important to understand the performance of the BEST ALIGNMENT algorithm first before adding in any of the rejection techniques. We want to find out if we can identify True Positives. Table 8.1 shows the accuracy of the BEST ALIGNMENT method with EDIT DISTANCE and the BEST ALIGNMENT method with NGRAMS. Table 8.1 also shows the same results when the NGRAMS linkage method is used at different levels of noise. The group-to-group linkage table has been included so that we can be reminded of the performance of the group to group method and compare it to the alignment approach. The NGRAMS group to group method performed better. It appears that the introduction of alignments did not improve performance. It was more difficult to correctly identify the real linkage between an $\mathbf{m}$ and a $\mathbf{b}$ with the dynamic method. However the results are still promising and heat maps generated from the BEST ALIGNMENT method appear in order.

Whilst this initial experiment used the EDIT DISTANCE 7.3.1 to measure $P(\mathbf{m}_i|\mathbf{b}_j)$ for the dynamic algorithm, in this thesis we will not continue further testing due to its poor performance. Intuitively when noise is set to zero, the accuracy should be 100% as for each comparison of a real match where both names are exactly the same the EDIT DISTANCE $= 0$. However this does not seem to be the case and the model is performing poorly. The problem with this model is that it is not inherently probabilistic so some arbitrary prior is needed to map EDIT DISTANCE scores to a probability that can be used in $P(\mathbf{m}_i|\mathbf{b}_j)$. This is risky as we have no prior this and used a best guess. We are more interested in using the NGRAMS to measure $P(\mathbf{m}_i|\mathbf{b}_j)$. This was because of its good performance in the models with no alignment and the good performance in Table 8.1.

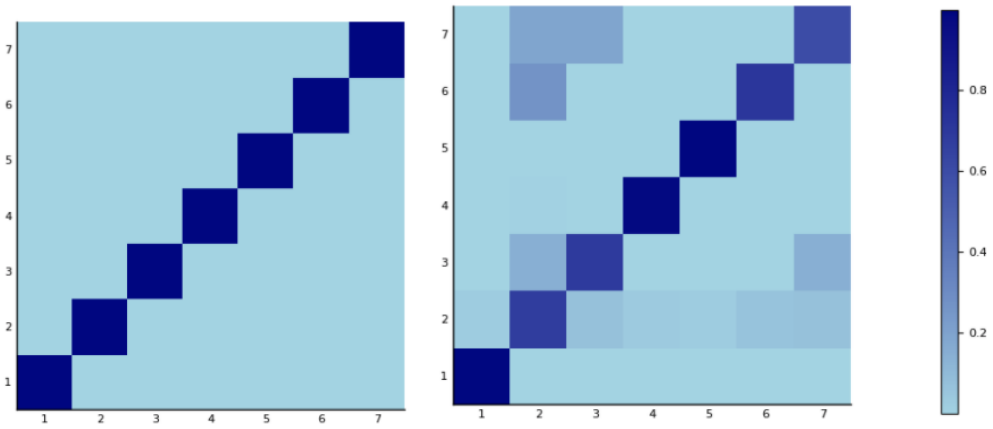


Figure 8.1: **Heatmaps on the `Baron` surname puzzle on test data with noise = 0.** Each row corresponds to a **b** group and each column corresponds to an **m** group. A dark blue square represents a high probability of the comparison being a real linkage and a light blue square represents a low probability of the comparison being a real linkage. Both graphs are on test data and the diagonal bands indicate that we can identify the true source of each family with both methods. *Left:* The heatmap generated when running the `group-to-group` with `ngrams` linkage method. *Right:* The heatmap generated when running the `best alignment` with `ngrams` linkage method. The diagonal is not as clear as in the `group-to-group` with `ngrams` linkage method.

8.2 Linkage Results including rejections

This section outlines the use of the rejection methods from Chapter 5 with the BEST ALIGNMENT linkage method. Two of the rejection methods used in the alignment agnostic method were used on the BEST ALIGNMENT method as well. These are ENTROPY and BTN. Again all of the following tests were done on the test data set explained earlier in section 3.2.1.

8.2.1 ENTROPY

Figure 8.2 shows the entropy of each row of the results grid as shown in previous figures. As before the entropy from True Negative comparisons are in red and the True Positives are shown in green. The green points are sorted by their entropy value and the red points represents the same comparison but with the family of interest replaced by a random family (so we know rejection should have happened).

There is not an obvious place to put an ENTROPY rejection threshold as the red and green data points are not linearly separable. We cannot tell which are the Positive results and which are the negative results from entropy. This is especially interesting when compared to other entropy results, such as the 6.4 where there almost all of the red entropy values are very low and grouped together at the bottom of the graph. In the BEST ALIGNMENT results there is a very wide range of entropy values for both positive and negative comparisons. This means it is difficult to differential the real matches from the fake. When noise is introduced in figure 8.3, the red circles extend down and spread out more. This would make entropy even less effective. This is a problem as we do believe that the real data is noisy.

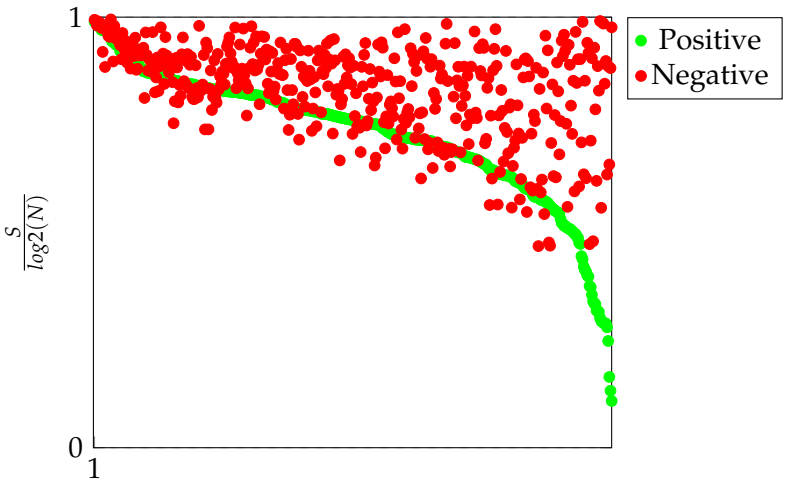


Figure 8.2: **Entropy for each family in the test data under the the BEST ALIGNMENT linkage method using NGRAMS without noise.** The red data points are typically above their corresponding data points. This suggests that setting a threshold on entropy is not particularly useful as the positive (green) points have a large range and there is no way to linely seperate the red and the green points.

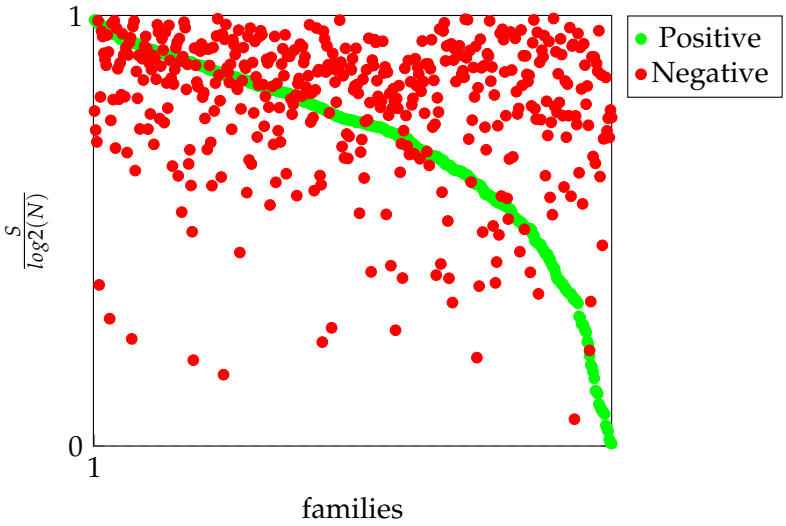


Figure 8.3: **Entropy for each family in the test data under the the BEST ALIGNMENT with noise = 0.2.** Here all of the data points have lower entropy than when there was no noise, both green and red. The spread makes it even more difficult to place a threshold as there are many low entropy red points below the majority of green ones. The red and green points have the same range.

	BEST ALIGN & EDIT DIST		BEST ALIGN & NGRAMS		NGRAMS group-to-group	
Noise	True +	False -	True +	False -	True +	False -
0.0	0.590	0.410	1.000	0.000	1.000	0.000
0.1	0.381	0.619	0.857	0.143	1.000	0.000
0.2	0.278	0.722	0.714	0.286	1.000	0.000
0.3	0.203	0.797	0.714	0.286	0.857	0.143
0.4	0.181	0.819	0.571	0.429	0.857	0.143
0.5	0.149	0.851	0.286	0.714	0.286	0.714
0.6	0.137	0.863	0.286	0.714	0.429	0.571
0.7	–	–	0.000	1.000	0.429	0.571
0.8	–	–	0.143	0.857	0.000	1.000
0.9	–	–	0.000	1.000	0.000	1.000
1.0	–	–	0.000	1.000	0.000	1.000

Table 8.1: **Accuracy of linkage methods without rejection at different noise values**. *Left:* The performance of the BEST ALIGNMENT method (using EDIT DISTANCE) by itself is poor. It is unable to correctly link more than 60% of linkages even when noise is set to zero. *Center:* Performance of the BEST ALIGNMENT using an NGRAMS, by itself is very good (100% of families correctly linked at zero noise). Performance is still very good even when some noise is introduced. *Right:* Performance of the NGRAMS group-to-group linkage method. This performs much better than the BEST ALIGNMENT method. There is a higher true positive rate at higher noise levels than the BEST ALIGNMENT method.

8.2.2 BTN Rejection

The BTN rejection method did not perform well at all on the BEST ALIGNMENT results. Table 8.4 shows this in the low precision and accuracy values. We can also see it in Figure 8.4, where in both the left and right histograms there is no significant difference between the green (positive match) and red (no match).

8.3 Summary

When the ENTROPY threshold rejection method is used with the BEST ALIGNMENT linkage method, we do not see promising results. Whilst positive results with high entropy tend to have an associated negative result with higher entropy there is no separation between the two groups. The BTN rejection method also performs poorly when used in conjunction with the BEST ALIGNMENT linkage method, and it is unable to distinguish between positive and negative test examples.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.1	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.2	0.008	0.992	0.997	0.003	0.733	0.502	0.015
0.3	0.024	0.976	1.000	0.000	1.000	0.512	0.046
0.4	0.048	0.952	0.998	0.002	0.953	0.523	0.091
0.5	0.075	0.925	0.990	0.010	0.886	0.533	0.138
0.6	0.165	0.835	0.966	0.034	0.828	0.566	0.276
0.7	0.333	0.667	0.876	0.124	0.729	0.605	0.458
0.8	0.624	0.376	0.698	0.302	0.674	**0.661**	0.648
0.9	0.914	0.086	0.316	0.684	0.572	0.615	0.704
1.0	0.987	0.013	0.000	1.000	0.497	0.494	0.661

Table 8.2: **Performance of ENTROPY rejection method on test data, using the BEST ALIGNMENT - NGRAMS linkage method with noise = 0.** Threshold is done against $entropy/log(2, n)$. This table shows that the ENTROPY rejection method is not performing well on the dynamic data. The highest accuracy value is 0.661 at a threshold of 0.8. This is considerably lower than when entropy was used with N-grams which had accuracy as high as 0.9 on the same test data set with the same conditions of β and n.

Threshold	True +	False -	True -	False +	Precision	Accuracy	F1
0.0	0.000	1.000	1.000	0.000	NaN	0.500	0.000
0.1	0.021	0.979	1.000	0.000	1.000	0.510	0.041
0.2	0.039	0.961	0.997	0.003	0.933	0.518	0.074
0.3	0.061	0.939	0.984	0.016	0.794	0.523	0.113
0.4	0.082	0.918	0.975	0.025	0.765	0.528	0.148
0.5	0.136	0.864	0.967	0.033	0.806	0.552	0.233
0.6	0.229	0.771	0.906	0.094	0.709	0.568	0.347
0.7	0.329	0.671	0.793	0.207	0.614	0.561	0.428
0.8	0.556	0.444	0.648	0.352	0.612	**0.602**	0.583
0.9	0.736	0.264	0.272	0.728	0.503	0.504	0.597
1.0	0.854	0.146	0.000	1.000	0.461	0.427	0.598

Table 8.3: **Performance of ENTROPY rejection method on test data, using the BEST ALIGNMENT - NGRAMS linkage method with noise = 0.2.** Threshold is done against $entropy/log(2, n)$. This table shows that the ENTROPY rejection method is not performing well on the dynamic data. The highest accuracy value is 0.602 at a threshold of 0.8. This is the same threshold that performed the best in Table 8.3. This is considerably lower than when entropy was used with N-grams which had accuracy as high as 0.818 on the same test data set with the same conditions of β and n.

Noise	True +	False -	True -	False +	Precision	Accuracy	F1
0.000	0.987	0.013	0.000	1.000	0.497	0.494	0.661
0.2	0.829	0.171	0.000	1.000	0.453	0.414	0.586

Table 8.4: **BEST ALIGNMENT linkage with BTN rejection on test data.**The BTN method seems to reject almost none of the comparisons. This is why the True positive rate is so high but simultaneously the True negative rate is zero.

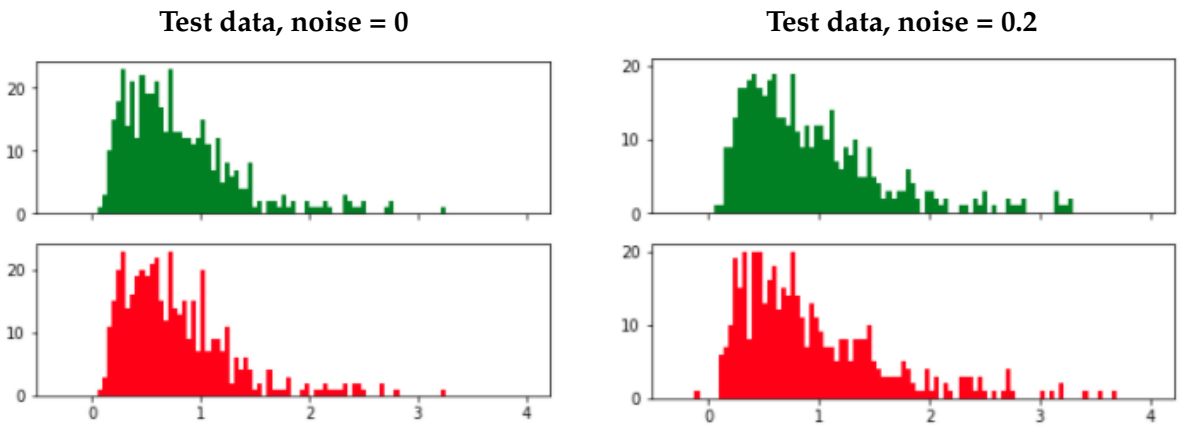


Figure 8.4: **BEST ALIGNMENT linkage with BTN rejection.** Histograms of scores using the BTN method in Section 5.4. The noisy environment (*right*) creates almost the same distribution as the one with no noise (*right*). Both positive and negative values seem to have the same performance of *Left*: The BEST ALIGNMENT linkage method with the BTN rejection method where no noise is applied to the test data. *Right:* The BEST ALIGNMENT linkage method with the BTN where noise=0.2 applied to the test data.

Chapter 9

Real data

This chapter details the results of the linkage finding and rejection algorithms on real Māori Land Online records. It was a stretch goal for the thesis to apply the methods to real data. As discussed previously, we are unable to measure the performance of the algorithms over the whole data set due to the lack of labelled data. Instead we investigate individual results that seem promising. The purpose of this section is to display and discuss examples from the results.

9.1 Māori Land Online

We look to close the loop by applying our models to the real Māori Land Online data. This will be a test two fold, firstly on what our models can find and secondly on how helpful the BDM-H data is in terms of being suitable to link to MLO.

The MLO data that was used in the test set was used, this means that the same surnames are being tested on the real data. All of the names in the following tests are present in both MLO and BDM. There are no surnames names missing from MLO, so it would be technically possible (while unlikely) to match every single family. The parameters $n = 4$ and $\beta = 0.5$ were used, these being guided by the results in earlier experiments. The noise method was only used to simulate the real noise in the MLO data set so is not needed for these experiments. A different base distribution was used in the NGRAMS model - one trained on the real MLO data. Last names were not included in the tests themselves as discussed earlier in Section 3.2.

Each of the models from previous chapters is applied to the real MLO data. We look at the results and investigate the rejections.

9.1.1 Group-to-group

This section focuses on different rejection methods (AL1, ENTROPY and BTN) applied to the group-to-group linkage methods discussed in Chapter 4. This section features output from the linkage/rejection methods which describes which linkages were rejected and the top three candidate **b** families (even if they were rejected).

9.1.1.1 AL1

When using the AL1 rejection method on the group to group linkage methods, there were vastly different results between the MATCH and NGRAMS methods, see Table 9.3. The AL1 rejection method requires a linkage to be *at least* as good as the worst scoring comparison when the **m** set is compared to members of itself. There were not many exact name matches,

so many of the linkages were rejected under the MATCH model. There were significantly fewer linkages rejected when using the NGRAMS model. This indicates there are families with similar distributions that are good enough to pass AL1 but not the same names (thus rejected by MATCH).

Linkage method	MATCH	NGRAMS
Rejected	426/1075	53/1075

Table 9.1: **Predicted MLO AL1 rejections with GROUP-TO-GROUP linkage method using MATCH and NGRAMS.** Just under half of the families are rejected with the MATCH linkage method and only 53 are rejected using the NGRAMS linkage method.

Let us compare the example linkage in Figure 9.1 of the same `Bracken` family groups which are matched using the NGRAMS linkage method instead of EXACT shown in Figure 9.2. Interestingly both families were rejected when the NGRAMS linkage method was used instead of the EXACT linkage method. Both methods also had different orderings and different top **b** families. The second **m** family still chose **b** families 2 and 6.

9.1.1.2 ENTROPY

We consider rejecting linkages that have entropy above some designated threshold. In the previous Chapters several different thresholds were tested so that we could understand how many linkages would be rejected at each threshold. The experiments in Chapters 6 and 8 suggested that very low entropy threshold values produced the best accuracy results. The same thresholds applied on the real MLO data results in almost all of the linkages rejected as shown below in Figures 9.3 and 9.4.

When using the ENTROPY rejection method on the group to group linkage methods, there were different results between the MATCH and NGRAMS. In the MATCH linkage method in Figure 9.3 there are clear and visible 'steps'. This is reflected in Table 9.7 where the number of rejections is not continuous. With the NGRAMS linkage model there was an even spread in entropy values which can be seen by the smooth diagonal in Figure 9.4.

Threshold	MATCH	NGRAMS
0	1075/1075	1075/1075
0.1	1066/1075	937/1075
0.2	1066/1075	863/1075
0.3	1066/1075	785/1075
0.4	1062/1075	661/1075
0.5	1054/1075	500/1075
0.6	1036/1075	366/1075
0.7	937/1075	236/1075
0.8	962/1075	96/1075
0.9	872/1075	18/1075
1.0	90/1075	0/1075

Table 9.2: **Predicted MLO ENTROPY rejections with GROUP-TO-GROUP linkage method using MATCH and NGRAMS.**. When the entropy threshold is set to the best value from earlier experiments, almost all of the families are rejected when using the NGRAMS linkage model. The MATCH linkage method also causes almost all of the families to be rejected at every threshold.

```
------------------------------------------------------------------
Method for linkage = match
Method = AL1
lastName =  Bracken
---------------Results------------------
M family: [ pakoa,  pakoa,  henare,  rameka,  joseph,  kawariki ]

All BDM families were rejected.

 position  id  BDM-H family
    1      1   [ hope, norman, maud, may, olive, annie, hilda, alice, robert,
               james ]
    2      2   [ jane, alexander, lilian, mary, helen, angel, roberta, henry,
               elizabeth, christopher, john, hugh, richard, maggie, christina,
               sarah ]
    3      3   [ john, elizabeth, lucy, fanny ]
 - - - - - - - - - -
M family: [ victor,  ngaire,  joseph,  anthony, lei,  duncan, matariki,
john,  charles,  douglas,  huia,  leilani,  kahakaha,  rachel, tui,
oileen,  mei,  ann ]


 position  id  BDM-H family
    1      2   [ jane, alexander, lilian, mary, helen, angel, roberta, henry,
               elizabeth, christopher, john, hugh, richard, maggie, christina,
               sarah ]
    2     3    [ john, elizabeth, lucy, fanny ]
    3      6   [ rose, clarence, annie, agnes, vivian, john ]
------------------------------------------------------------------
```

Figure 9.1: **The `Bracken` family, linkage selected using BTN and the MATCH linkage method.** In this example with the `Bracken` family, two of the Bracken MLO sets are shown with the best found linkages. In the first, the linkage is rejected and the model says that no families were suitable. There are no names that appear in the **m** family and in any of the top 3 performing possible **b** families. It is good that these were rejected. The second family was not rejected. Because there is *at least one* exact match in each of the top three **b** families. All of the **b** families match on the name `john`. Both of the MLO families in the example chose BDM family 2 in their top 3.

```
----------------------------------------------------------------------
Method for linkage = ngrams
Method = AL1
lastName =  Bracken
---------------Results------------------
M family: [ henare,  joseph,  pakoa,  rameka,  kawariki,  pakoa ]

All BDM families were rejected.

 position  id  BDM-H family
    1      7   [ vera, annie, norman, margaret, kathleen, imelda, charles ]
    2      6   [ clarence, vivian, agnes, john, rose, annie ]
    3      5   [ nolan, hamilton, mary, hazel, william, walr, stanley, ira,
               maud, eileen, olive ]

 - - - - - - - - - - -
M family: [ oileen,  john,  matariki,  leilani,  kahakaha,  anthony ,
charles,  huia,  tui,  mei,  duncan,  ann,  victor,  joseph,  ngaire,
lei,  douglas,  rachel ]

All BDM families were rejected.

 position  id  BDM-H family
    1      7   [ vera, annie, norman, margaret, kathleen, imelda, charles ]
    2      2   [ richard, angel, alexander, maggie, sarah, roberta, helen,
               christopher, henry, john, mary, elizabeth, lilian, christina,
               jane, hugh ]
    3      6   [ clarence, vivian, agnes, john, rose, annie ]
----------------------------------------------------------------------
```

Figure 9.2: **The `Bracken` family, linkage selected using BTN and the NGRAMS linkage method.** In this example with the `Bracken` family, two of the Bracken MLO sets are shown with the best found linkages. In the first, the linkage is rejected and the model says that no families were suitable. There are no names that appear in the **m** family and in any of the top 3 performing possible B families. It is appropriate that these were rejected. The second family was also rejected, even though there is an exact match in each of the second and third **b** families.

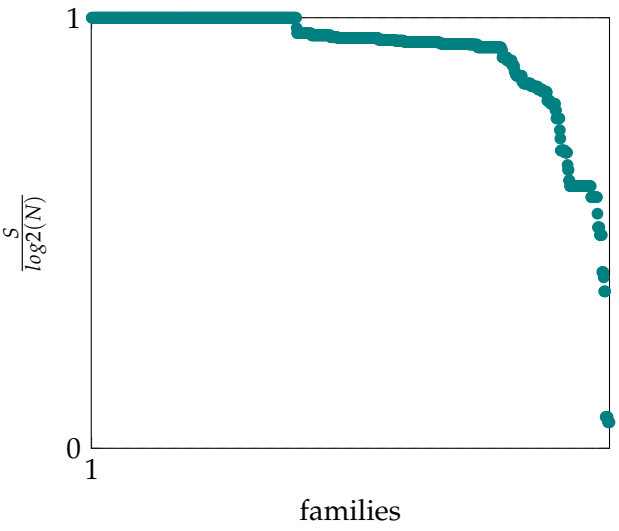


Figure 9.3: **Sorted entropy for each family in the MLO data set when compared to the families in BDM-H using NGRAMS**. There is a wide range of entropy value with a few clear steps in the distribution. There are many values with an entropy of 1, another portion with almost 1. There are 2 further clusters, at 0.6 and 0.1. The obvious steps in the distribution suggest that some families were very clearly present and others clearly not.

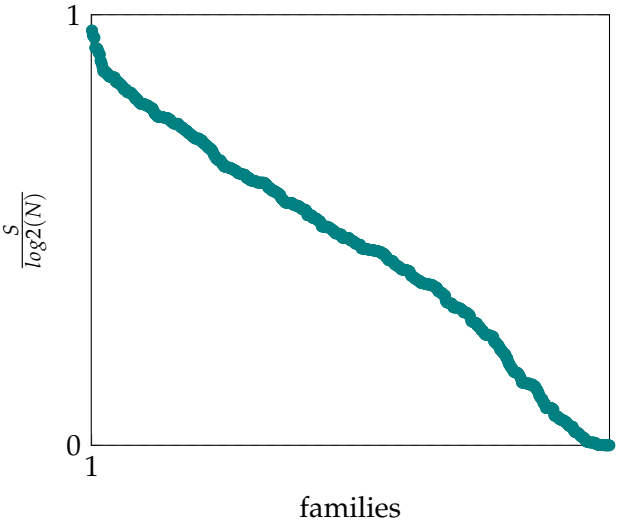


Figure 9.4: **Sorted ENTROPY using NGRAMS instead of MATCH.** Each data point represents an MLO to BDM-H comparison. There is a wide range of entropy values, and there are no clear and significant steps when the data points are sorted by entropy. There are no obvious steps or clusters that we can use to set an appropriate threshold. Instead because there is a continuous distribution of entropy we can only make a best guess and set a sensible threshold based on examples and performance on test data.

```
--------------------------------------------------------------------
Method for linkage = ngrams
Method = Entropy (0.1)

---------------Results------------------
lastName =  Smillie
------------------------------

M family: [jullianne, julieanne, julianne, malcolm, akuira]

 position  id  BDM-H family
    1      1   [isabella, millar, alexander, daniel, john, julians ]
    2      6   [helen, scott, cowan, william ]
    3      8   [william, charles, alexander, mcpherson, james, jessie , per,
               robert, constance, isabell, john, melville ]
--------------------------------------------------------------------
```

Figure 9.5: **The `Smillie` family, linkage selected using ENTROPY and the NGRAMS linkage method.** There is only a partial match `julians` to `julianne`, however due to what looks like an error the name `julianne` is repeated 3 times so it becomes more compelling. Examples like this make a good case for alignment specific methods because that would not allow `julianne` to count more than once.

In the examples below a threshold of 0.1 was used as this value had the highest accuracy in Chapter 6. There are some comparisons that were not rejected but probably should have been on further inspection.

There was one family that appears to be a genuine match. There are lots of identical name matches and a few partial matches. The `Hollamby` family are the best example of a probable linkage in this data set, an example of how accurate the linkage appears to be can be seen in Figure 9.7.

9.1.1.3 BTN

When the BTN rejection method is applied to NGRAMS linkages on the MLO data we can see from Figure 9.8 that the shape and positioning of the real scores (shown in blue), closely resembles the shape and positioning of the negative cases (shown in red) reprinted from Figure 6.6. The majority of the scores from the real data (and the test negatives) are less than zero. A score less than zero means that **b** set was worse at generating **m** than the base distribution was. This suggests that there are not many (if any) real matches between MLO and BDM-H.

9.1.2 Linkages found with best alignment

9.1.2.1 BTN

The BTN rejection method with the NGRAMS-DYNAMIC linkage method only rejected 196 families when the threshold was set to zero. This is of a similar order to the BTN with NGRAMS using the group-to-group method instead of the alignment with dynamic. This can be seen in Table 9.4.

```
------------------------------------------------------------------
Method for linkage = ngrams
Method = Entropy (0.1)
---------------Results-----------------
lastname = Meager
---------------------------------------
M family: [ allan,  rotorua,  florence,  ngaire,  john,  robert ]

 position  id  BDM-H family
    1      1   [ thomas, edward, john, william, florence, mary ]
    2      3   [ elizabeth, mary, jane, ka, john, alice, harriet, ellen, ann,
               clara, william ]
    3      7    [ alfred, john, annie, elizabeth, catherine, mary ]
------------------------------------------------------------------
```

Figure 9.6: **The `Meager` family, linkage selected using ENTROPY and the NGRAMS linkage method.** There are two matches here - `florence` and `john`. This seems like a reasonably compelling link but there are a lot of people missing from both sides.

```
------------------------------------------------------------------
Method for linkage = ngrams
Method = Entropy (0.1)
---------------Results-----------------
lastName =  Hollamby
-------------
M family:[ esme,  b,  cecil, ngaire,  ann,  kenneth,  yvonne,  john ,
albert,  joan,  mavis,  rewa ]


 position  id  BDM-H family
    1      2   [ esme, alberta, carita, william, kenneth, howard, cecil, allan,
               ngaire, annie, allan, jack, doris, eileen ]
    2      3   [ phyliss, beryl, albert, edward, mavis, rewa, flora,
               dardanella, joan, margaret, moyra ]
    3      5   [ flora, may, james, roderick, richard, albert ]
------------------------------------------------------------------
```

Figure 9.7: **The `Hollamby` family, linkage selected using ENTROPY and the NGRAMS linkage method.** In this linkage the `Hollamby` family has identical matches with the 'best' match : `Esme, Cecil, Ngaire, Kenneth` and a partial match `albert` which is very similar to `alberta`. There is also identical matches with the second best match: `Albert, Mavis, Rewa and Joan`. A possible explanation for this is that the MLO group is actually made up of cousins (which would explain why they have the same share value), having all inherited from a grandparent. So we would have a family made up of : `Esme, Cecil, Ngaire and Kenneth` and then another one with `Albert, Mavis, Rewa and Joan`. `Yvonne` and `john` might be other family members.

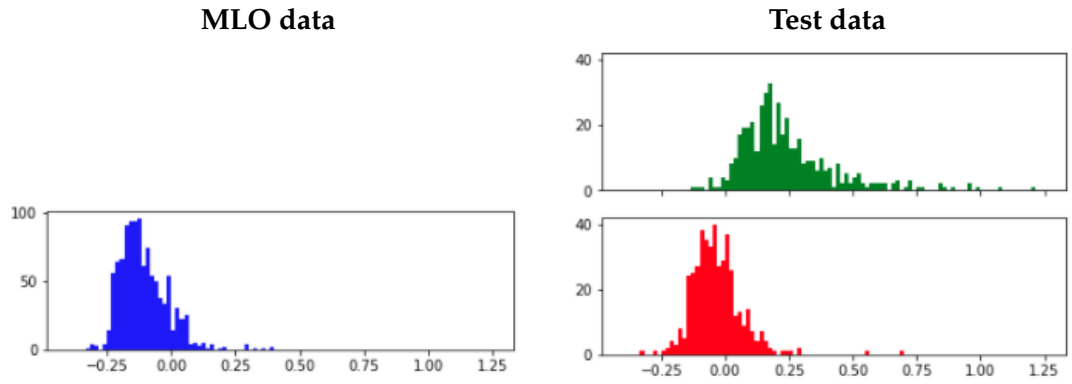


Figure 9.8: **NGRAMS linkage with BTN rejection.** *Left:* Histogram of scores using the BTN method 5.4. The BTN rejection method with n = 4 and beta = 0.5. The majority of the scores are less than zero. Because most of the scores are less than zero this means that the base distribution was better at generating the target **m** than **b** was. *Right:* histogram scores for BTN with NGRAMS on test data with noise. This graph is included for reference.

```
-----------------------------------------------------------------------
Method for linkage = ngrams
Method = BTN
    ---------------Results------------------
lastName =  Wesley
---------------------------------------
M family:[ marie,  morehu,  kaye,  emily,  iwi,  leonie,  ,  anne,  mary ]

 position  id  BDM-H family
    1      3   [ eliza, annie, emily ]
    2      5   [ william, john, lawrence, thelma, isabell ]
    3      4   [ william, richard, john ]

 - - - - - - - - - -
M family:[ tahatu,  alethea,  john,  charles,  hohepa,  may ]

All BDM families were rejected.

 position  id  BDM-H family
    1      2   [ john, arthur, edward, charles ]
    2      1   [ walr, charles, john ]
    3      7   [ edith, john, hellen ]

-----------------------------------------------------------------------
```

Figure 9.9: **The `Wesley` family, linkage selected using BTN and the NGRAMS linkage method.** In the first wesley family linkage, the result was not rejected. There was one exact match in `emily` and a partial match in `anne` to `annie`. In the second linkage, all of the BDM-H families were rejected. This is unusual as there are two exact matches `john` and `charles`.

Threshold	NGRAMS
-0.2	146/1075
-0.1	648/1075
0.0	951/1075
0.1	1051/1075
0.2	1065/1075
0.3	1071/1075
0.4	1075/1075
0.5	1075/1075
0.6	1075/1075
0.7	1075/1075
0.8	1075/1075
0.9	1075/1075
1.0	1075/1075

Table 9.3: **Predicted MLO BTN rejections with GROUP-TO-GROUP linkage method using NGRAMS**.This table shows the number of rejections at each threshold value when using the BTN rejection method. Many families were rejected when using the NGRAMS linkage method, when the threshold is set to zero there are 951 families rejected. This means that for 951 familes the base distribution was a better generator of the target **m** set than any **b** family ngram frequency distributions.

Figure 9.10 shows the BTN rejection histograms with the NGRAMS-DYNAMIC linkage method.

9.1.2.2 ENTROPY

The ENTROPY rejection method rejected every linkage at the low entropy thresholds (where we would have set them based on our experiments on test data). This can be seen in Figure 9.11 where a significant number of the data points are close to one at the top of the graph.

The results from the Table 9.5 suggest that there are not many individuals from MLO in BDM-H. This is evident because of the high number of rejections at the best entropy thresholds from the test data. There are low entropy comparisons, but there are not a significant number of them. It is unlikely that both data sets were generated from the same ground truth. There are a few convincing matches and these stand out. However the majority of the low entropy comparisons are not so obvious. They typically have one or two identical matches and a large number of non matches. The size of these families means that the there are lots of names going into the NGRAMS distribution. This can mean that by chance the **m** and the **b** families produce similar enough NGRAM substring probabilities to be counted as a match.

9.1.3 Summary of results on MLO

From the results of the tables and looking at the comparisons there is very little evidence that there are more than (at best) a handful of real matches between BDM-H and MLO. Because BDM-H is a historic data set, there are no recent records freely available. This means it is almost impossible to find any modern day shareholders who have lost contact using BDM-H. This is unfortunate and highlights the difficulty of the task that Māori land incorporations face. They are looking for a very small needle in a very large haystack.

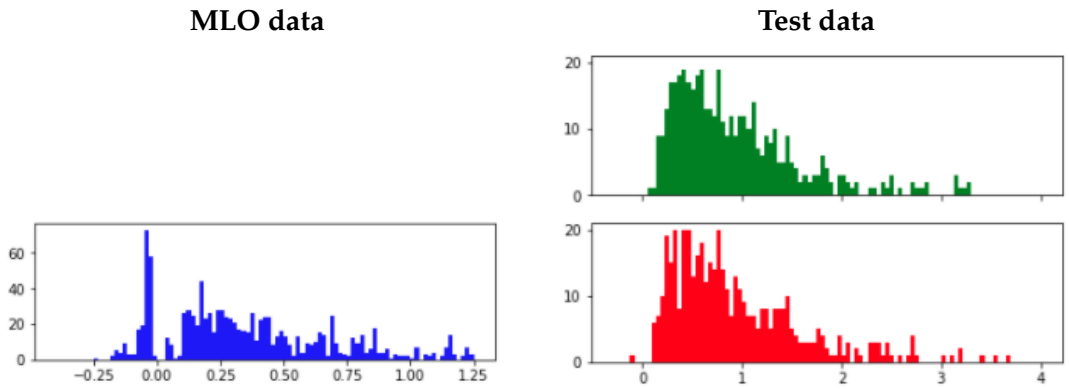


Figure 9.10: **NGRAMS-DYNAMIC linkage with BTN rejection.** *Left:* Histogram of scores using the BTN method 5.4 on MLO data. The BTN rejection method with n = 4 and beta = 0.5. The histogram is very spread out and not bell shaped like the other BTN histograms, it has a large peak at just below zero and then spreads out from just above zero up to 1.25. Because most of the scores are more than zero this means that **b** was better at generating the target **m** than the base distribution was. *Right*:This is the same BTN model with NGRAMS-DYNAMIC with noise = 0.2 applied to the test data. This graph is included for reference.

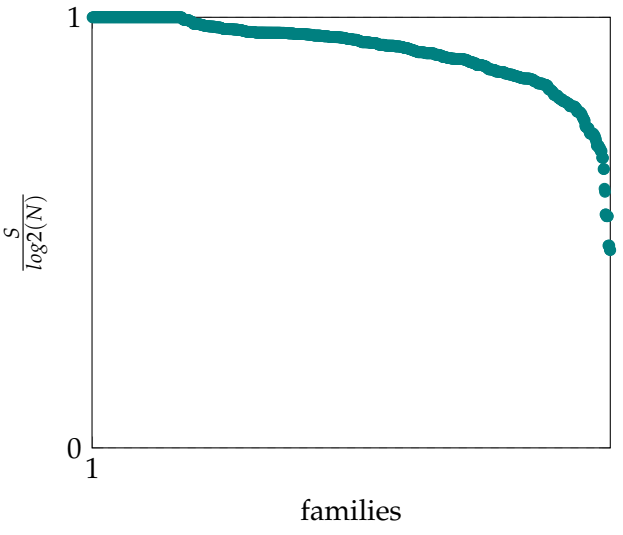


Figure 9.11: **Sorted entropy for each family in the MLO dataset when compared to the families in BDM-H using the BEST ALIGNMENT NGRAMS linkage method.** There is a smaller range in values for this linkage method, most of the entropy values are near one and the lowest value is around 0.45.There are no obvious steps in the distribution.

Threshold	NGRAMS
-0.2	1/1075
-0.1	24/1075
0.0	196/1075
0.1	219/1075
0.2	384/1075
0.3	528/1075
0.4	634/1075
0.5	741/1075
0.6	790/1075
0.7	864/1075
0.8	903/1075
0.9	953/1075
1.0	969/1075
1.1	1012/1075
1.1	1026/1075

Table 9.4: **BTN rejections with NGRAMS-DYNAMIC alignments on MLO data**. When the **btn** threshold is set to the best value from earlier experiments, almost none of the families are rejected when using the NGRAMS linkage model. There were 196 families rejected at threshold = 0.0, which would suggest that there are some reasonable matches between MLO and BDM-H.

9.2 Application of models: Cenotaph Data

Records from the online cenotaph offer an opportunity to test the linkage and rejection methods on another data set. An Auckland War Memorial Museum project, the Online Cenotaph stores details for more than 235,000 New Zealand service men and women, who have served this country on active service from the 19th century until today. [69]

The purpose of including this additional data set is that it is another opportunity to test the group comparison method against Births Deaths and Marriages. The data is much cleaner and smaller, and has been maintained by the Auckland War Memorial Museum project. As a part of the project, relatives of service men and women are able to add photographs and additional details to the profiles of their loved ones.

The limitation of the data that we were able to retrieve is that it is only contains males and only where there were more than one family member who fought. So essentially the data set is a collection of families of brothers who went to war. This still allows for some interesting comparisons to BDM as we can still assess the outcomes of the models. We may get higher entropy than we would normally expect for a perfect match as there are so many other names in the BDM sets that are not in the Cenotaph.

In *theory* this should perfectly line up with bits of BDM-H as most of this data is from before the 100 year cutoff.

The challenge with this data set is that (like with MLO) we do not have the answers, so we can only run our models and then go through and assess the results. Even then it is hard to say what the correct linkages are - and therefore we can only guess at which methods are most effective.

There are 20 surnames that are in our Cenotaph data that are not present in our Birth Deaths and Marriages data set.

Threshold	NGRAMS
0	1075/1075
0.1	1075/1075
0.2	1075/1075
0.3	1075/1075
0.4	1075/1075
0.5	1070/1075
0.6	1063/1075
0.7	1051/1075
0.8	986/1075
0.9	779/1075
1.0	7/1075

Table 9.5: **Predicted MLO rejections with BEST ALIGNMENT linkage using NGRAMS**. When the entropy threshold is set to the best value from earlier experiments, all of the families are rejected when using the NGRAMS linkage model.

9.2.1 Group-to-group linkage methods

This section focuses on different rejection methods (AL1, ENTROPY and BTN) applied to the group-to-group linkage methods discussed in Chapter 4.

9.2.1.1 AL1

Table 9.8 contains the number of linkages that would be rejected for both the MATCH and NGRAMS linkage method when using the AL1 rejection method on the Cenotaph data. With both of the linkage methods there were very few families rejected by the AL1 method. This means that in almost all of the cases there were at least one string that appeared in both the **b** set and in the CEN family (**c**) set. Less than 10% of families were rejected with both linkage methods.

Predicted AL1 Rejections with various linkage methods on Cenotaph data		
Linkage method	MATCH	NGRAMS
Rejected	40/735	61/735

Table 9.6: **Predicted Cenotaph AL1 rejections with GROUP-TO-GROUP linkage method using NGRAMS**. This table shows the number of rejections for each of the different linkage methods when using the AL1 rejection method. Both of the linkage methods reject very few of the families. This suggests that many of the Cenotaph families are present in the BDM historical data.

Figure 9.12 shows an example linkage from the Cenotaph data when using the AL1 rejection method with the MATCH linkage method. It performs very well as the whole of the Cenotaph family is present in the most likely linkage. This is a relatively small example and it true that family *31* has many other names in it. This is likely to be siblings who did not go to war.

Figure 9.13 displays the results for when the NGRAMS method is used in conjunction with al1 instead. This is another example of a realistic looking match where all of the cenotaph individuals are present in the most likely family (76).

```
------------------------------------------------------------------
Method for linkage = match
Method for rejection =  AL1
    ---------------Results------------------
lastName =  Childs
---------------------------------------
C family: [ charles ,  ellis ]

 position  id  BDM-H family
    1      31  [ albert , arthur , edith , john , ellis , ellen , charles ]
    2      14  [ charles , vine , valentine , newlands , eliza , james ,
               mclachlan , hector , william , george , nellie , raymond , mary
               , annie , archibald ]
    3       5  [ joseph , jessie , clara , charles , ellen , warden ]
------------------------------------------------------------------
```

Figure 9.12: **The `Childs` family, linkage selected using AL1 and the MATCH linkage method.** In this example with the `Childs` family, the best found linkages are shown. The BDM-H families was not rejected. This is because there is *at least one* exact match in each of the top three **b** families. All of the names are present in the top BDM candidate. There are two exact matches in the first **b** family (`Charles` and `Ellis`), and one in the other two (Charles).

```
------------------------------------------------------------------
Method for linkage = ngrams
Method for rejection =  AL1
---------------Results------------------
lastName =  Davies
-------------
C family:  [ percy ,  james ,  arthur ,  sylvesr ]

 position   id   BDM-H family
    1      76    [ lonsdale , arthur , percy , wesrfield , james , ethel ,
                 albert , harry , silvesr , annie , william , george , rita ,
                 thomas , elsie , allan ]
    2     157    [ henry , david , lesr , william ]
    3      93    [ arthur , david , bertie , victor , darcy , james , william ]
------------------------------------------------------------------
```

Figure 9.13: **The `Davies` family, linkage selected using AL1 and the NGRAMS linkage method.** In this example with the `Davies` family, the best found linkages a shown. The BDM-H families was not rejected. This is because there is *at least one* exact match in each of the top three **b** families. All of the names are present in the top BDM candidate. The really impressive thing is the change of spelling of `Sylvesr/silvesr`. It is nice to see that an example with noise is still picked up (albeit with a lot of other perfect names). We can also see that there are a few additional names in BDM so this suggests some other siblings who were in the family.

9.2.1.2 ENTROPY

We consider rejecting linkages that have entropy above some designated threshold. In the previous Chapters several different thresholds were tested so that we could understand how many linkages would be rejected at each threshold. The experiments in Chapters 6 and 8 suggested that very low entropy threshold values produced the best accuracy results. The same thresholds applied on the real MLO data results in almost all of the linkages rejected as shown in the previous Section in Figures 9.3 and 9.4. When these same thresholds were applied to the Cenotaph data entropy results appear very similar to the real MLO application, this can be seen in Figures 9.14 and 9.15 and most of the linkages were rejected using the previously successful threshold value of 0.1. This is interesting because when we look at the actual examples such as is shown in Figure 9.16 and 9.17 the actual linkages are much different in that they appear legitimate. Almost all of the cenotaph families have an realistic looking match.

When using the ENTROPY rejection method on the group to group linkage methods, there were similar results between the MATCH and NGRAMS. In the MATCH linkage method in Figure 9.3 the distribution is continuous and concave up. This is reflected in Table 9.10 where the number of rejections consistent increasing with each threshold value. With the NGRAMS linkage model there was an even spread in entropy values which can be seen by the smooth diagonal in Figure 9.4.

Predicted ENTROPY Rejections with various linkage methods on Cenotaph data		
Threshold	MATCH	NGRAMS
0	715/735	715/735
0.1	697/735	560/735
0.2	678/735	499/735
0.3	642/735	457/735
0.4	619/735	400/735
0.5	597/735	344/735
0.6	561/735	284/735
0.7	516/735	212/735
0.8	426/735	107/735
0.9	279/735	38/735
1.0	20/735	20/735

Table 9.7: **Predicted Cenotaph ENTROPY rejections with GROUP-TO-GROUP linkage method using MATCH and NGRAMS.** This table shows the number of rejections for each of the different linkage methods when using the ENTROPY rejection method. When the entropy threshold is set to 0.1 which was the best value from earlier experiments, almost all of the families are rejected when using the NGRAMS linkage method. The ENTROPY on the MATCH linkage method is the most aggressive, rejecting the more families at low threshold values than the NGRAMS methods. Note that there were 20 surnames that were present in the Cenetaph data but were missing from BDM-H. All of these were (of course) rejected, but this explains why the rejection values at the highest ENTROPY threshold was not 100%

In Figure 9.16 both names from the Cenotaph family were present in the best performing BDM-H family but it was still rejected. This suggests that additional names in the **b** family that do not match the **m** or in this case **c** family detract from the plausibility of the linkage. This is a flaw in the entropy rejection method as it is not designed to model plausibility as a human would see it.

There are a few interesting examples where the exact property that NGRAMS was picked

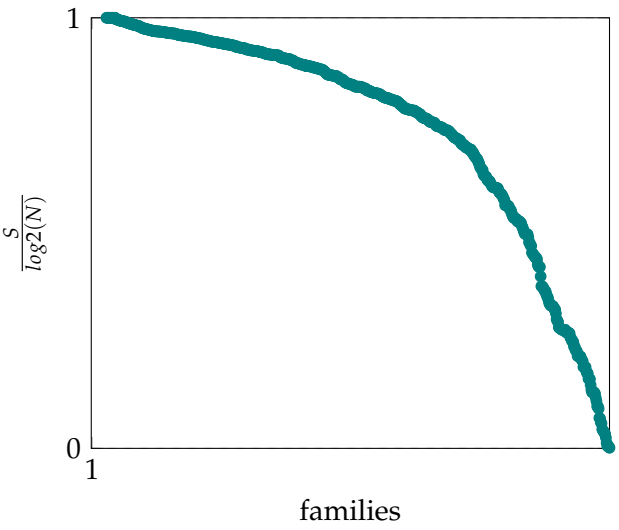


Figure 9.14: **Sorted entropy for each family in the Cenotaph data set when compared to the families in BDM historical using the MATCH linkage method**. There is a big range of entropy values, and they follow a continuous curve with no steps or clusters. The shape is concave down, decreasing. This means that there are in general, more high entropy values.

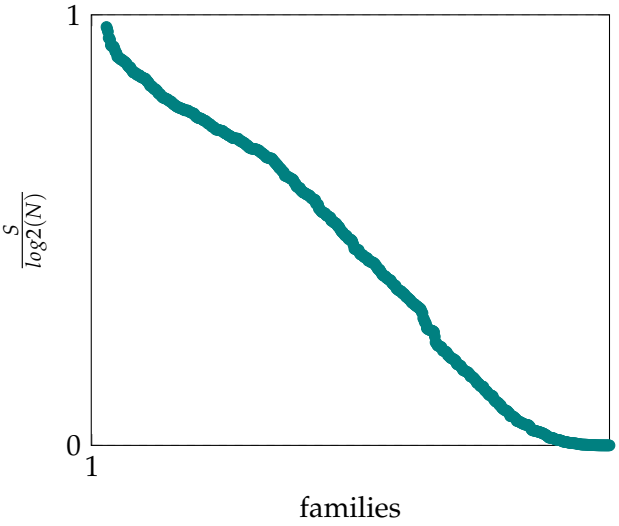


Figure 9.15: **Sorted entropy for each family in the Cenotaph dataset when compared to the families in BDM-H using the NGRAMS linkage method**. There is a big range of entropy values, and they follow a continuous curve with no steps or clusters. The shape is linear, decreasing. This means that there are in general, the same amount of high entropy values as low entropy values.

```
--------------------------------------------------------------------
Method for linkage = match
Method for rejection = entropy (0.1)
---------------Results------------------
lastName =  McCathie
----------------------------------------
C family:  [ james ,  david ]

All BDM families were rejected.

 position   id   BDM-H family
    1       1    [ stuart , william , david , james , elspeth , henderson ,
                 henderson , jane , keith , hall , donald , margiet , malcolm
                 ]
    2       2     [ thelma , conville , jean , cecil , raymond , keith , william
                 , donald ]
    3       3     [ colin , donald , thomas , stuart ]
--------------------------------------------------------------------
```

Figure 9.16: **The `McCathie` family, linkage selected using ENTROPY and the MATCH linkage method.** In this example with the `McCathie` family, the best found linkages a shown. The BDM-H families were rejected. Strangely, all of the Cenotaph names are present in the top BDM candidate. We can also see that there are a few additional names in BDM so this suggests some other siblings who were in the family. This suggests that ENTROPY is not as able to cope with the concept of having a small number of perfect matches like AL1 and BTN rejection methods are.

```
-----------------------------------------------------------------
Method for linkage = ngrams
Method for rejection = entropy
--------------Results-----------------
lastName =  McLean
---------------------------------------
C family:  [ kenzie ,  alexander ,  john ]

 position   id   BDM-H family
    1       22   [ james , john , mclachlan , janet , david , henry , anew ,
                 duncan , helen , sarah , mckenzie , alexander , hugh , isabella
                 ]
    2       10   [ ann , alexander , mary , john , john , annie , kenzie , jessie
                 , ann , murdoch , lillie , donald , glasgow , catherine , swart
                 , duncan , ann , evangeline ]
    3      334   [ william , james , bell , john , alexander , elizabeth , anew ,
                 george ]
-----------------------------------------------------------------
```

Figure 9.17: **The `McCathie` family, linkage selected using ENTROPY and the MATCH linkage method.** In this example with the `McCathie` family, the best found linkages a shown. The BDM-H families were not rejected. The top candidate has two exact matches and one partial match in the top BDM candidate. `alexander` and `john` match exactly and then we also have `kenzie` and the partial match `Mckenzie`. We can also see that there are a few additional names in BDM so this suggests some other siblings who were in the family.

to solve was seen - spelling differences were picked up. In the Figure 9.17 for example the name `kenzie` can be partially matched to `Mckenzie`.

9.2.1.3 BTN

The **match** linkage method is not applicable to the BTN because it does not use the base distribution to generate names. It is only applicable on the NGRAMS linkage method. When the BTN rejection method is applied to NGRAMS linkages on the Cenotaph data we can see from Figure 9.18 that the shape and positioning of the real scores (shown in blue), is more similar to the shape and positioning of the positive cases (shown in green) where the peak of the scores are above zero. The majority of the scores from the real data (and the test positives) are more than zero. A score more than zero means that **b** set was better at generating **m** than the base distribution was. This suggests that there are potentially many real matches between the cenotaph data and BDM-H.

With the BTN method (when the threshold was set to 0) there were also very few families rejected.

In Figure 9.19 there are only two BDM-H families that have the last name `Caffery` so only these are displayed in the results. This is an example of a linkage that looks realistic and has two exact matches but is rejected anyway. This is likely because of the large family size, so that the base distribution is better at predicting the other 14 names than the **b** set is.

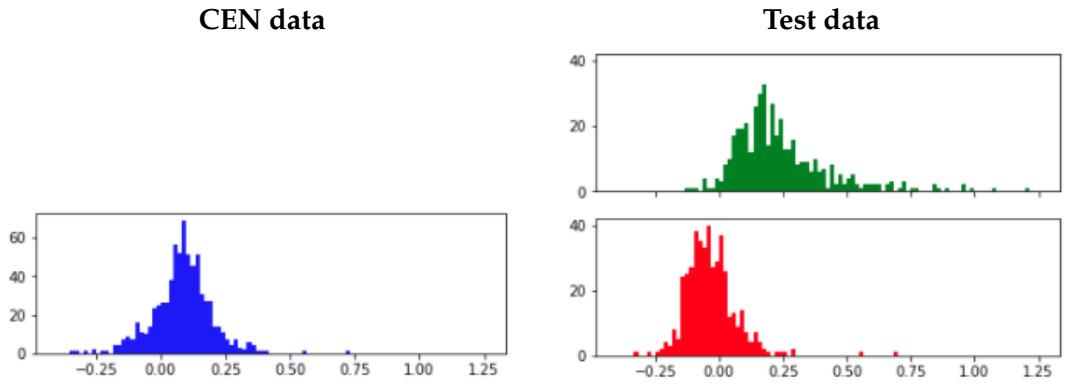


Figure 9.18: **NGRAMS linkage with BTN rejection.** *Left:* Histogram of scores using the BTN method 5.4. The BTN rejection method with n = 4 and beta = 0.5. The majority of the scores are more than zero. A score more than zero means that the **b** set was better at generating **m** than the base distribution was. *Right:* histogram scores for BTN with NGRAMS on test data with noise. This graph is included for reference.

```
----------------------------------------------------------------------
Method for linkage = ngrams
Method for rejection = BTN
---------------Results------------------
lastName =  Caffery
---------------------------------------
C family:  [ james ,  joseph ,  william ]

All BDM families were rejected.


 position  id  BDM-H family
    1      1   [ edith , may , mary , jane , joseph , maria , elizabeth ,
               donald , henry , margaret , albert , mona , ann , rose , robert
               , james ]
    2      2    [ ernest , gibson , george , joseph , stanley ]
----------------------------------------------------------------------
```

Figure 9.19: **The `Caffery` family, linkage selected using ENTROPY and the MATCH linkage method.** In this example with the `Caffery` family, the best found linkages a shown. The BDM-H families were rejected. The top candidate has two exact matches. `james` and `joseph` match exactly, but `william` was not present. Again we can also see that there are a few additional names in BDM so this suggests some other siblings who were in the family.

Threshold	NGRAMS
-0.2	20/725
-0.1	60/735
0.0	159/735
0.1	426/735
0.2	658/735
0.3	715/735
0.4	732/735
0.5	733/735
0.6	734/735
0.7	734/735
0.8	735/735
0.9	735/735
1.0	735/735

Table 9.8: **Predicted Cenotaph BTN rejections with GROUP-TO-GROUP linkage method using NGRAMS.**This table shows the number of rejections at each threshold value when using the BTN rejection method. When the threshold was zero, not many families were rejected when using the NGRAMS linkage method. This means that for 159 familes the base distribution was a better generator of the target **m** set than any **b** family ngram frequency distributions.

9.2.2 Linkages found with best alignment

9.2.2.1 BTN

The BTN rejection method with the NGRAMS-DYNAMIC linkage method rejected zero families when the threshold was set to zero. This is less rejections than when the BTN rejection method was used with NGRAMS using the group-to-group method instead of the alignment with dynamic. This can be seen in Table 9.10, and Figure 9.20 where all of the scores are more than zero.

Figure 9.22 shows the performance of the BTN rejection method in blue. The distribution is bell-shaped with a tail on the right hand side. This means that in every linkage comparison the **b** family was a better predictor of the Cenotaph set than the base distribution.

9.2.2.2 ENTROPY

In previous Chapters different thresholds were tested so that we could understand how many linkages would be rejected at each threshold. The experiments in Chapters 6 and 8 suggested that very low entropy threshold values produced the best accuracy results. The same thresholds applied on the real CEN data results in almost all of the linkages rejected when ENTROPY rejection is used with the BEST ALIGNMENT linkage method. The ENTROPY rejection method rejected every linkage at the low entropy thresholds (where we would have set them based on our experiments on test data). This can be seen in Figure 9.22 where a significant number of the data points are at the top of the graph. Table 9.10 shows the exact number of linkages rejected at every threshold value.

9.3 Summary of methods

Across the MLO data, all of the methods performed similarly. Almost all linkages were rejected. This was reflected in the example linkages shown throughout Section 9.1 where

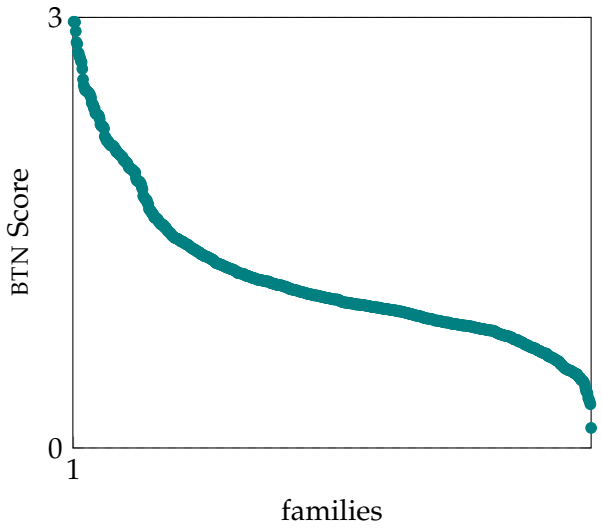


Figure 9.20: **Sorted BTN score for each family in the Cenotaph data set when compared to the families in BDM historical using the NGRAMS linkage method**. There is a large range of BTN values, from 0.14 to 3. There is a continuous curve with no steps or clusters. Most of the data points are less than 1.5.

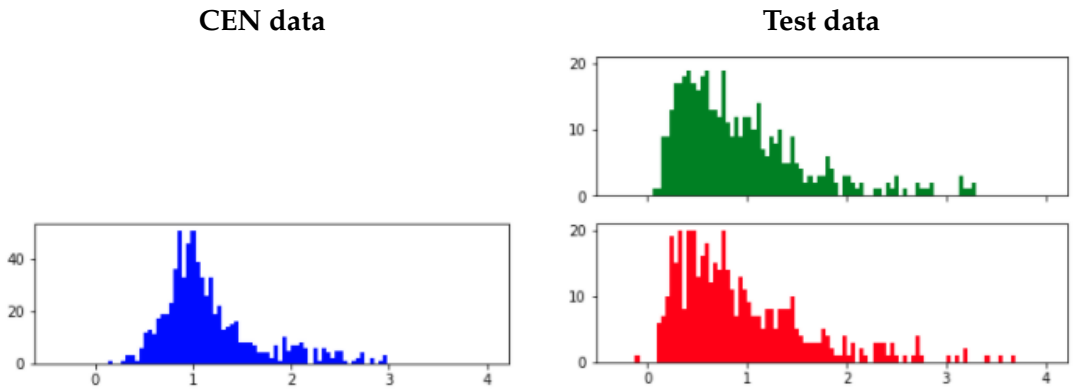


Figure 9.21: **NGRAMS-DYNAMIC linkage with BTN rejection.** *Left:* Histogram of scores using the BTN method 5.4. The BTN rejection method with n = 4 and beta = 0.5. None of the scores are less than zero, this means that the base distribution was better at generating the target **m** than **b** was. *Right*:This model has noise = 0.2 applied to the test data. This graph is included for reference.

Threshold	NGRAMS
0	0/735
0.2	1/735
0.4	8/735
0.6	44/735
0.8	129/735
1.0	325/735
1.2	477/735
1.4	553/735
1.6	600/735
1.8	619/735
2.0	644/735
2.2	672/735
2.4	689/735
2.6	702/735
2.8	709/735
3.0	715/735

Table 9.9: **Predicted Cenotaph BTN rejections with BEST ALIGNMENT linkage method using NGRAMS**. When the BTN threshold is set to the best value from earlier experiments (0.5), almost none of the families are rejected when using the NGRAMS linkage model. The MATCH linkage method also causes almost all of the families to be rejected at every threshold.

many did not resemble each other at all. There were a few believe able matches that we came across but not many.

The Cenotaph data on the other hand provided much more interesting and differing results. The AL1 and BTN methods rejected almost none of the linkages and found many promising results while the ENTROPY method rejected everyone. The ENTROPY method on the Cenotaph data had a similar distribution of entropy values to the real MLO data set. This is surprising as the actual examples in the cenotaph data are much more promising.

In general a problem with the ENTROPY rejection method is that it is hard to set the threshold. We have tried out a variety of different thresholds but we have no real way of justifying where to put it.

What the AL1 and BTN methods have in common is that they are both accepting linkages that show some promise. This is particularly useful in the Cenotaph data where (we believe) the whole family would not go to war and so we would expect to see only a partial match in names between the BDM-H set and the Cenotaph set.

A common theme for those families picked in the examples is that when no really obvious matches exist, very large families are picked instead. This could be a fault of the linkage methods in that a big family is just more likely to *by chance* have the names from the MLO/CEN family.

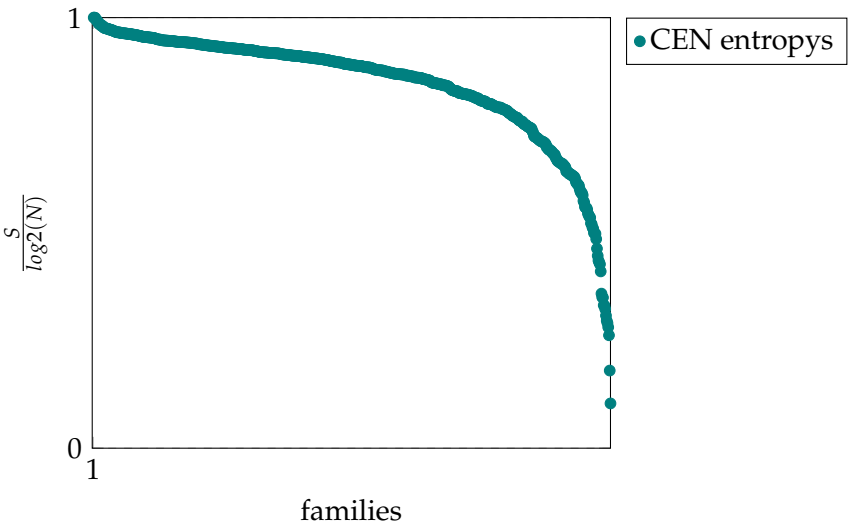


Figure 9.22: **Sorted entropy for each family in the CEN dataset when compared to the families in BDM historical using the BEST ALIGNMENT linkage method with NGRAMS.** There is a much more defined slope in the graph compared to the equivalent display in Figure 9.11. Most of the entropy values are above 0.75 but there are some that are lower. The lowest value is around 0.1. There are no values less than 0.1 which is unfortunate as this was where the entropy values tended to be set on test data.

Threshold	NGRAMS
0.0	735/735
0.1	735/735
0.2	733/735
0.3	729/735
0.4	722/735
0.5	612/735
0.6	694/735
0.7	649/735
0.8	571/735
0.9	361/735
1.0	20/735

Table 9.10: **Predicted Cenotaph ENTROPY rejections with BEST ALIGNMENTS linkage method using NGRAMS.** When the entropy threshold is set to the best value from earlier experiments, almost all of the families are rejected when using the NGRAMS linkage model. The MATCH linkage method also causes almost all of the families to be rejected at every threshold.

9.4 Discussion of real data

Can we use these methods to reliably find individuals from Māori Land Online in Births Deaths and Marriages (historical) ?
This is the million dollar question (well approximately 5 million dollar question for PKW). The short answer is no. There were very few matches that appear legitimate from the results of the linkage and rejection methods on the real MLO to BDM-H comparison. However many compelling matches were produced when the linkage and rejection methods were applied to the cenotaph data. When the MLO results are compared to the results of the CEN data, the reasons for the lack of matches is highlighted. It appears that the methods themselves perform well, because CEN produces a lot of compelling results. This implies that it is a problem with the data quality in MLO. There are two main problems with the comparison that we are trying to do.

The first is that MLO is vastly out of date. Because the MLO data set is only updated in response to an inheritance event, this relies on individual shareholders to take their inheritance claims to the Māori Land Court. The onus is on the shareholders to follow complex processes for low reward. This cannot be guaranteed to happen and requires not insignificant effort from the individual.

The cenotaph is a reliable data set because we know that the family groupings in it are fairly accurate as they have been updated by historians and family members. With the CEN data it is likely that both data sets were generated from the same latent entity i.e. a single individual created the record in BDM and in CEN.

The second problem is that MLO and BDM-H do not cover the same time period. One of the key differences between the MLO and CEN data sets is the time period of the data. As a result of the fact that MLO does not always contain the most recent information, for any given land block it is possible that some of the original owners are still listed. MLO data references many different points during New Zealand's history. This is unlike the CEN data which is close to representing singular events in time (New Zealand has not fought in many large scale wars). We know that many of the current day owners in MLO are too young to even appear in BDM-H. This suggests that a high number of rejections is the result of using BDM-H as compared to what we would expect from a more up to date BDM.

One of the goals of the wider project has been to investigate how much could be done with the freely available data and build a case for requesting more complete data be made available (by the Crown for example). It is apparent that further (current) BDM data is required, this would mean that any shareholder from MLO would be in theory present in BDM. It is very likely that if BDM was used instead of the historical version then there would be more matches as there were in the cenotaph data which matches the time frame of BDM-H.

Chapter 10

Conclusion

Part of what has made this research journey different to a traditional thesis is that right from the very beginning it has been grounded in an understanding of the problem in the context of te ao Māori and the world of Māori data. This has changed the way that the project worked as a whole and it has been a privilege to be a part of. It has meant changing our expectations and learning how to integrate Māori data and information with a western view of how science is carried out. It is important to bear that in mind when reflecting on the scope and the directions that this research has taken.

10.1 Key contributions:

This thesis has modelled the key inference processes in the BDM and MLO linkage problem using a Bayesian model. This has involved understanding and integrating te ao Māori into the research problem. By having a Māori world view at the forefront of this research it has meant that we have taken into consideration inherent cultural structures that are present within the data. Māori concepts such as whānau, hapū, iwi and whakapapa are crucial to understanding how land ownership and land use are seen by Māori. This has directly led to the concept of GROUP-TO-GROUP comparisons and the effort to link groups of people to each other, as opposed to linking individual records independent of their context. By understanding the company that an individual keeps, we are closer to understanding them and their story.

We explored two different GROUP-TO-GROUP linkage methods, which we called MATCH and NGRAMS. The MATCH method assigned scores to each possible linkage via the number of exact string matches between the two sets. This performed well on test data but, unsurprisingly, was not robust to noise. The NGRAMS linkage method used a Markov model for text to identify the likelihood of one group creating the other. This was much more robust to noise and outperformed the MATCH method on test data.

Three novel rejection methods for linkages were designed: AL1, ENTROPY and BTN. The AL1 method rejects linkages that perform worse than an on-the-fly threshold, which is created by considering a single perfect match. When using the MATCH linkage method, this means rejecting all linkages where there is not *at least one* individual who appears in both sets. When using the NGRAMS linkage method which is probabilistic this method rejects linkages that do not do at least as well as a set with one identical match. The ENTROPY rejection method uses the (scaled) Shannon entropy [63] to measure the distribution of normalised posterior probabilities derived from scores across different possible linkages in a puzzle. We proposed that low entropy indicates strong winners and so one could reject all linkages with high entropy. This method performed well on test data for GROUP-TO-GROUP

but did not do as well on the BEST ALIGNMENT linkage methods, or on real data. The BTN method rejects linkages where the base distribution is better able to generate the target **m** (or **c**) set than the proposed **b** set.

A dynamic programming algorithm was developed for generating the BEST ALIGNMENT between an **m** and a **b** set. The BEST ALIGNMENT linkage method was implemented with two different string matching methods: NGRAMS and EDIT DISTANCE. Of these, NGRAMS was an extension of the model used in the GROUP-TO-GROUP linkage and the EDIT DISTANCE approach measures the number of changes needed to be made to turn one string into the other. The BEST ALIGNMENT approach was not as successful on test or real data.

These linkage methods were then applied to real MLO and Cenotaph data. We were able to identify many compelling matches between the Cenotaph data and BDM-H. We were unfortunately not able to identify many compelling matches between the MLO data and BDM-H.

10.2 Assessment of data

Part of this project was to assess the value of having this data. The key questions were:

- *Is Births Deaths and Marriages (historical), good enough?*
- *Would it be worth having real BDM?*

The results of the various experiments on both the real Māori Land Online data and the Cenotaph data indicate that Births Deaths and Marriages Historical is not sufficient. Cenotaph had many positive linkages that could be identified as a very likely link. This suggests that our methods perform well enough to identify real links that are present. By contrast, the MLO data did not yield convincing matches. We hypothesise that whilst there may be issues with the identification of sibling groups in MLO, the main problem is that MLO data is misaligned with BDM-H. There is a not enough overlap in terms of the time that the individuals are alive.

It would be the recommendation from this thesis that for the project to proceed further and be able to provide valuable insight to Māori land organisations, access to Births Deaths and Marriages be sought.

10.3 Future research and work

As with any research project, there is always more that could be done. Key areas of further work would be:

10.3.1 Build an interface for this recommendation engine to be used by PKW

Currently this research remains purely experimental and has not been implemented in a way that is immediately consumable by PKW. This is likely to be done as a part of the wider project but would require investigation into real use cases and the exact queries that PKW staff would be making.

10.3.2 Rejection methods more appropriate for the BEST ALIGNMENT linkage method

It was demonstrated that the ENTROPY and BTN rejection methods did not perform very well in combinations with the BEST ALIGNMENT linkage method. It would be useful to

further develop rejection methods that utilised specific features of the BEST ALIGNMENT linkage method. These are not limited to but may include use of the number of *perfect* matches within an alignment, and the number of matches within an alignment that are not to a real person (i.e. the number of times the best alignment occurs when there is no actual alignment).

10.3.3 Estimation via Importance Sampling

The BEST ALIGNMENT algorithm finds the optimal alignment and uses the probability derived from this as the basis for the subsequent comparison between $\mathbf{b}$ groups. From a Bayesian perspective this is not the best thing to do, as it uses $f = p(\mathbf{m} \mid \mathbf{b}, \mathbf{z}^\star)$, with $\mathbf{z}^\star$ being the optimal alignment, as a proxy for $F = p(\mathbf{m} \mid \mathbf{b})$.

An alternative approach to this is to use a form of Importance Sampling ([70, 68]) to generate a Monte Carlo estimate of the (latter) *marginal likelihood* - an approach that integrates over the options for alignment rather than optimising over them. Use of Importance Sampling to estimate F potentially offers two advantages over the optimisation approach of using f. Firstly, it directly addresses the correct probability and this means it takes appropriate account of ambiguous identities, rather than just "taking the best match". Secondly, the computational load involved is readily tuned up or down as needed: more computational resource can generate better approximations when uncertainty is large, or can be cut down when the evidence is clear.

To estimate F with Importance Sampling, we would like to draw samples $\mathbf{z}$ from an optimal proposal distribution q, which is known to be $q(\mathbf{z}) = |f(\mathbf{z})|\ p(\mathbf{z})$. Such samples could be generated using Metropolis Hastings for example. Although investigated in initial experiments there was not time to fully develop this as a solution.

10.3.4 Investigate use of Latent Dirichlet Allocation (LDA) at a land block level

"LDA is a probabilistic model with a corresponding generative process – each document is assumed to be generated by this (simple) process" [71]. LDA (often called "topic modelling") models sets of documents, each of which is taken to consist of words from more than one topic. A topic is simply a distribution over a fixed vocabulary, and it is assumed that these topics precede the creation of the documents [71]. We could consider the names on a land block as a document. Some initial work was done investigating use of LDA to generate hapū groups across different share blocks and minute book references, using this mixed membership model to try and get at which names "travel together" as a whole in the MLO corpus.

There were some promising initial results that were able to group different well known families together, albeit with only a modest number of topics being assumed. Although this line of investigation was not continued here, it may be worth pursuing further. It would be interesting to see whether the LDA model could recreate family groups at a level of granularity that approaches what was achieved by the minute book references and share values.

To close

This has been more than a scientific journey but also one of cultural discovery. Understanding the underlying problem of Māori who have become disconnected from their culture has shown me things about my own cultural connection (or lack thereof). As a half Cantonese-Chinese, half English, New Zealander, I too am disconnected from my land. I, like many other New Zealand-born Chinese, am disconnected from my language. This has prompted me to change my perspective of who I am and where I belong.

This project has totally changed my perspective of New Zealand history and the land. It has also allowed me to enrich my own cultural connections as I realised that I too am lost from my culture and I am working to get that back.

Bibliography

[1] Palin, T. W. *Sketch map of the North Island of New Zealand shewing native tribal boundaries, topographical features, confiscated lands, military and police stations, etc. 1869.* Retrieved from https://kura.aucklandlibraries.govt.nz/digital/collection/maps/id/710, New Zealand Defence Office, Wellington (1869)

[2] Science For Technological Innovation *Analytics to identify and connect successors to whenua* Retrieved on June 1st, 2020 from: https://www.sftichallenge.govt.nz/our-research/projects/spearhead/analytics-to-identify-and-connect-successors-to-whenua/

[3] Parininihi ki Waitotara *Website* Retrieved from *https* : //*pkw.co.nz*/ (22/08/18).

[4] Ministry for Culture and Heritage *New Zealand's 19th-century wars* Retrieved 28th May 2020 from https://nzhistory.govt.nz/war/new-zealands-19th-century-wars/introduction updated 22-Aug-2017.

[5] Boast, R. *Te tango whenua – Māori land alienation - Raupatu – confiscations* Retrieved from http://www.TeAra.govt.nz/en/te-tango-whenua-maori-land-alienation/page-4 te ara - july, 2015.

[6] te ara *whakapapa-genealogy.* Retrieved from https://teara.govt.nz/en/whakapapa-genealogy (29/08/18).

[7] Cram, F. *Rangahau Māori: Tona tika, tona pono - The validity and integrity of Māori research.* In Research Ethics in Aotearoa New Zealand (pp. 35–52). Auckland: Pearson Education. (2001).

[8] *Māori Dictionary* Retrieved from: https://maoridictionary.co.nz/search?idiom=phrase=*proverb=loan=histLoanWords=keywords=matauranga+maori*

[9] *Māori Dictionary* Retrieved from: *https://maoridictionary.co.nz/search?idiom=phrase=proverb=loan=histLoanWords=keywords=Whanaungatanga*

[10] *Māori Dictionary* Retrieved from: https://maoridictionary.co.nz/search?idiom=phrase=*proverb=loan=histLoanWords=keywords=manaakitanga*

[11] *Māori Dictionary* Retrieved from: http://www.katoa.net.nz/kaupapa-maori/kaupapa-*maori-research-ethics*

[12] *Māori Dictionary* Retrieved from: https://maoridictionary.co.nz/word/3424

[13] *Maori Ethical Frameworks* Retrieved from: http://www.rangahau.co.nz/ethics/166/

[14] Joanne Clapcott, Erica Williams, Anne-Marie Jackson, Daniel Hikuroa, Chris Hepburn, Jamie Ataria and Rauru Kirikiri *Blending mātauranga Māori and Western science* Retrieved from: https://www.royalsociety.org.nz/news/blending-matauranga-maori-and-western-science/

[15] Henrik Moller *Matauranga Maori, science and seabirds in New Zealand* 36:3, 203-210, DOI: 10.1080/03014220909510151 Retrieved from: https://doi.org/10.1080/03014220909510151

[16] Kennedy, V., Wehipeihana, N. *A stock take of national and international ethical guidelines on health and disability research in relation to Indigenous People (Unpublished Report).* The National Ethics Advisory Committee Te Kahui Matatika o te Motu, Wellington. (2006).

[17] Fyers, Andy and Hartevelt, John *Nā Niu Tīreni / New Zealand Made* Retrieved from https://interactives.stuff.co.nz/2018/07/na-niu-tireni-new-zealand-made/ (2020-02-09)

[18] Fellegi, I., Sunteb, A. *A theory for record linkage* Dominion Bureau of Statistics and The American Journal of Statistics Vol 4 (1969).

[19] Steorts, R. Ventura, L., Sadinle, M., Fienberg, S. *A Comparison of Blocking Methods for Record Linkage* International Conference on Privacy in Statistical Databases: Privacy in Statistical Databases pp 253-268(2014)

[20] Rebbeca Steorts. *Entity Resolution with empirically motivated priors* (2015)

[21] Tancredi, Steorts, R. *A unified framework for deduplication and population size estimate* (2019)

[22] Peter Christen. *A Two-Step Classification Approach to Unsupervised Record Linkage* Department of Computer Science, The Australian National University, 2007

[23] National INstitute of Standards and Technology. *Manhattan distanceA Two-Step Classification Approach to Unsupervised Record Linkage* Retrieved 18/06/2020 from : https://xlinux.nist.gov/dads/HTML/manhattanDistance.html

[24] Peter Christen. *Febrl: a freely available record linkage system with a graphical user interface* HDKM '08: Proceedings of the second Australasian workshop on Health data and knowledge management - Volume 80January 2008 Pages 17–25

[25] Malmi, E., Gionis, A., Solin, A. *Computationally Inferred Genealogical Networks Uncover Long-Term Trends in Assortative Mating* Retrieved on July 15th from : https://arxiv.org/abs/1802.06055 submitted:16 Feb 2018

[26] Mauricio Sadinle and Stephen E. Fienberg. *A Generalized Fellegi-Sunter Framework for Multiple Record Linkage With Application to Homicide Record Systems* (2013)

[27] Stringham, T. *Fast Bayesian Record Linkage With Record-Specific Disagreement Parameters* arXiv: Methodology, Mathematics, Economics (2020)

[28] Christen, P. & Goiser, K. *Quality and complexity measures for data linkage and deduplication* Quality Measures in Data Mining', Springer Studies in Computational Intelligence, vol. 43, pp. 127–151 (2007).

[29] Andrieu, C., De Freitas, N., Doucet, A., & Jordan, M. *An Introduction to MCMC for Machine Learning* Machine Learning, 50, 5–43. (2003)

[30] Mohri, M., Rostamizadeh, A., Talwalkar, A. *Foundations of Machine Learning* The MIT Press (2012)

[31] Zhu, X., Andrew Goldberg, A. *Introduction to Semi-Supervised Learning* 9781598295481, Morgan and Claypool. 2009

[32] Christen, P. *A Two-Step Classification Approach to Unsupervised Record Linkage.* Department of Computer Science, The Australian National University. (2007)

[33] *Births Deaths and Marriages.* Retrieved from https://www.bdmonline.dia.govt.nz/ (18/08/18).

[34] Stanford *Minimum Edit Distance* Retrieved from *https://web.stanford.edu/class/cs124 /lec/med.pdf* (26/09/18).

[35] Linux *Manhattan Distance* Retrieved from *https://xlinux.nist.gov/dads/HTML/ manhattanDistance.html* (18/10/18).

[36] MLOL *Maori Land Online* Retrieved from http://www.maorilandonline.govt.nz (27/09/18).

[37] LINZ.govt *Land Information New Zealand* Retrieved from https://www.linz.govt.nz/data/linz-data (27/09/18).

[38] Huelsenbeck, J., Ronquist, F. *MRBAYES: Bayesian inference of phylogenetic trees.* Bioinformatics Applications Note. Vol 17, no.8 (2001)

[39] Baxter, R., Christen, P. & Churches, T. *A comparison of fast blocking methods for record linkage.* ACM SIGKDD Workshop on Data Cleaning, Record Linkage and Object Consolidation', Washington DC, pp. 25–27 (2003).

[40] Cohen W.W., Ravikumar P. & Fienberg S.E. , *A comparison of string distance metrics for name matching task.* IJCAI-03 workshop on information integration on the Web' (IIWeb-03), Acapulco, pp. 73–78 (2003).

[41] Kliss, B., Alvey, W *Record Linkage Techniques - 1985. Proceedings of the workshop on exact matching methodologies in Arlington, Virginia, May 9-10* Internal Revenue Service Publication, Washington DC (1985).

[42] Elfeky, M., Verykios, V. and Elmagarmid, A. *TAILOR: A record Linkage Toolbox.* Proc. of the 18th Int Conference on Data Engineering (2002).

[43] Kingma, D., Rezende, Z., Mohamed, S., Welling, M. *Semi-supervised Learning with Deep Generative Models.* To appear in the proceedings of Neural Information Processing Systems (NIPS) (2014).

[44] Dong-gi Lee, Sangkuk Lee, Myungjun Kim, Hyunjung Shina. *Historical inference based on semi-supervised learning* Expert Systems With Applications 106 (2018) pg 121-131.

[45] Deep Ai *Semi Supervised Learning.* Retrieved from https://deepai.org/machine-learning-glossary-and-terms/semi-supervised-learning (27/10/2018).

[46] Sydney Shep, Marcus Frean, Rhys Owen, Rere-No-A-Rangi Pope, Pikihuia Reihana, Valerie Chan. *Indigenous frameworks for data-intensive humanities: recalibrating thepast through knowledge engineering and generative modelling* To be printed in: Journal of Data Mining and Digital Humanities ISSN 2416-5999, an open-access journal

[47] Baker, G. *Implications of low Maori participation in the census.* Retrieved from https://thespinoff.co.nz/atea/01-08-2018/has-the-2018-census-failed-maori/ (2018).

[48] Kennedy, V., & Wehipeihana, N. *A stock take of national and international ethical guidelines on health and disability research in relation to Indigenous People (Unpublished Report)* The National Ethics Advisory Committee Te Kahui Matatika o te Motu, Wellington.(2006).

[49] Kohavi, R., Provost, F. *Glossary of terms* Machine Learning, vol. 30, no. 2-3, pp. 271-274, 1998.

[50] Mitchell, T. *Machine Learning* McGraw Hill, (1997).

[51] Chan, V. *Using Genetic Programming to automate Heuristic design for the Dynamic Time-dependant Orienteering Problem.* Honours thesis (2017).

[52] Taonui, R. *'Tribal organisation - The significance of iwi and hapū', Te Ara - the Encyclopedia of New Zealand,* Retrieved accessed 24 November 2018 http://www.TeAra.govt.nz/en/tribal-organisation/page-1.

[53] The whakapapaclub *Maori Naming Conventions* Retrieved accessed 16 July 2020 http://www.whakapapaclub.nz/information/maori-naming-conventions/

[54] Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford (2009). *Introduction to Algorithms (3rd ed.)* MIT Press. ISBN 978-0-262-03384-8.

[55] Robert A. Wagner AND Michael J. Fischer . *The String-to-String Correction Problem*. Journal of the Association for Computing Machinery, Vol. 21, No. I, January 1974, pp. 168–173.

[56] Melissa Nightingale *Thousands of adults, children legally changing their names in New Zealand every year*. The New Zealand Herald, 2 March, 2018.

[57] Daniel Jurafsky, James H. Martin. *Speech and Language Processing, An introduction to Natural Language Processing, Computational Linguistics and Speech recognition* Posts and Telecom press. pg 132.

[58] Daniel Jurafsky, James H. Martin. *Speech and Language Processing, An introduction to Natural Language Processing, Computational Linguistics and Speech recognition* Posts and Telecom press. pg 16 of chapter 3 .

[59] Murphy, K. *Machine Learning: A probabilistic Perspective* pg.596. 2012.

[60] Katz, S. M *Estimation of probabilities from sparse data for the language model component of a speech recognizer.* IEEE Transactions on Acoustics, Speech, and Signal Processing, 35(3), 400–401.

[61] Jurafsky, D., Martin, J. *Speech and Language Processing.* Pg 13 of chapter 3.

[62] Thorsten Brants Ashok C. Popat Peng Xu Franz J. Och Jeffrey Dean *Large Language Models in Machine Translation* Google (2007)

[63] Shannon C. E. *A Mathematical Theory of Communication* Reprinted with corrections from The Bell System Technical Journal, Vol. 27, pp. 379–423, 623–656, July, October, (1948).

[64] Bellman, R *Dynamic programming* ISSN: 0036-8075 ; PMID: 17730601 Version:1 Science (New York, N.Y.), 01 July 1966, Vol.153(3731), pp.34-7

[65] Derczynski, L. *Complementarity, F-score, and NLP Evaluation* LREC 2016 pp 261-266

[66] van Der Hoek, John Elliott, Robert J *Introduction to Hidden Semi-Markov Models* Cambridge Core All Books (Cambridge University Press), 2018

[67] Shi Wanli, Wang Hongyong *An Approach for Stereo Matching Using Pair-wise Sequence Alignment Algorithm Based on Dynamic Programming*

2010 International Conference on Challenges in Environmental Science and Computer Engineering, March 2010, Vol.1, pp.511-514

[68] Lee, Peter *Bayesian Statistics: An Introduction, 4th Edition*, 2012

[69] Melissa Nightingale *A memorial resource for the nation*. Retrieved 10th April 2020 from *https://www.aucklandmuseum.com/war-memorial/online- cenotaph/about-online-cenotaph*

[70] Press, William H and Teukolsky, Saul A and Vetterling, William T and Flannery, Brian P *Numerical recipes 3rd edition: The art of scientific computing*, Cambridge university press (2007)

[71] Clark, S. *Topic Modelling and Latent Dirichlet Allocation* Retrieved accessed 24 November 2018 from *https://www.cl.cam.ac.uk/teaching/1213/L101/clark*$_{l}$*ectures/lect*7*.pdf*

Appendix A

414 Aotea MB 282-283 dated 17 March 2020

414 AOT 282-283

ORDER DETERMINING A LIFE INTEREST

Te Ture Whenua Māori Act 1993, Section 18(1)(a)

In the Māori Land Court
of New Zealand
Aotea District

IN THE MATTER of an application to determine the life interest of Robert Tamou

AT a sitting of the Court held at Whanganui on 17 March 2020 before Layne Ross Harvey, Judge

WHEREAS application has been filed by Andrew Clayton Robinson to determine the life interest held by Robert Tamou

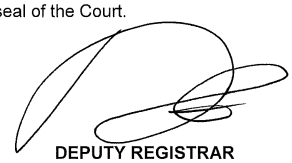
AND WHEREAS on 6 July 1995 at 50 Aotea MB 55-56 Robert Tamou was granted an interest for life or until remarriage in the estate of his late wife, May Bishop

AND WHEREAS Robert Tamou died on 30 November 2019

NOW THEREFORE the Court upon reading and hearing all evidence in support and being satisfied on all matters upon which it is required to be so satisfied HEREBY DETERMINES pursuant to section 18(1)(a) of Te Ture Whenua Māori Act 1993 the said life estate in favour of the persons whose names are set out in the attached schedule, in the proportions shown

AND pursuant to rule 7.5(2)(b) of the Māori Land Court Rules 2011 this order shall issue IMMEDIATELY

AS WITNESS the hand of a Deputy Registrar and seal of the Court.



DEPUTY REGISTRAR

A20200001684

2.

Aotea District

Blocks	CT Ref	Current Owner	Shares
Araukuku B	279126	Robert Tamou	0.0004
		Robert Tamou	0.0031
Motumate Block	459789	Robert Tamou	0.1111
Ngatikahumate 1M2	306981	Robert Tamou	7.3859
		Robert Tamou	1.0552
Ngatitamarongo No. 6	421351	Robert Tamou	0.0875
		Robert Tamou	0.0125
Ngatitara 26B	397461	Robert Tamou	61.85625
Ngatitara 6F No. 2	413258	Robert Tamou	0.025
Otamare	TN137/75	Robert Tamou	0.1111
Otukaia Grant 3808	482153	Robert Tamou	0.00103
Paora-Aneti 17 & 18	491965	Robert Tamou	0.0156
		Robert Tamou	0.0022
Part Araukuku F	TNE2/1219, TNE2/1220	Robert Tamou	0.0031
		Robert Tamou	0.0005
Part Town Belt Ohawe Township	TN13/87	Robert Tamou	0.0031
		Robert Tamou	0.0004
Section 69 Block VI Waimate District	TN7/201	Robert Tamou	0.1111
Tipoka Sec 55A	501706	Robert Tamou	0.0001
		Robert Tamou	0.000014
Waiokura A	TNH2/1210	Robert Tamou	0.1111
Wairua	TN10/56	Robert Tamou	0.00005
		Robert Tamou	0.000008
Waitaraiti Section 56 Block XII Cape SD	TN10/70	Robert Tamou	0.0001
		Robert Tamou	0.000015
Parininihi ki Waitotara Incorporation		Robert Tamou	393.737

Beneficiaries/Successors

No	Name	Sex	Address	Proportion
1	Andrew Clayton Robinson	M	32 Patu-Kukupa Street, Manaia 4612	¼
2	Daniel Shayne Robinson	MD		¼
3	Taina Shaun Tamou	M	21 Ribbonwood Street, Sippy Downs 4556, Queensland, Australia	¼
4	Joval Margaret Tamou	F		¼

A20200001684