Scotland's Census
Shaping our future
A' dealbhadh ar n-àm ri teachd

Scotland's Census 2022

Name Reordering Methodology

June 2020

National Records of Scotland

Contents

1. Plain English Abstract

On paper census forms there is a household section, followed by individual forms for each person. The household section includes a question on how the people in the household are related to each other. We want to know how the people who appear on the individual forms are related to each other. Therefore we need to correctly match up the individual forms to the people in the relationship question.

Usually the people in the household form will appear in the same order as the people in the individual forms. However, sometimes they do not. Therefore we plan to compare the names on the household form to the names on the individual forms, to make sure we match them up correctly.

2. Abstract

On the census returns almost all information relating to individuals is captured on the individual forms. The exception is information on the relationships between individuals, which is captured on a relationship matrix on the household form. In order to use the relationship information it must be attached to the information from the individual forms. In 2011 this was done by assuming that the respondents followed the guidance to enter persons on the individual forms in the same order as they appeared in the relationship matrix. Looking at the names that were entered, this was usually, but not always, the case.

The proposal here is to take the names from the various parts of the form and compare them. This can then be used to reorder the names, along with the corresponding information, on part of the form, so that the information matches up correctly. The comparison is done by measuring the similarity of the names on the individual form to each of the names on the relationship matrix and to each of the names entered at the start of the household form. The similarity scores take into account nicknames, phonetically similar names, names that agree at the start or the end, and also according to a character by character comparison. This information is then used to find the ordering that minimizes discrepancies. In some cases it will be obvious enough that the data could be reordered to the suggested ordering

automatically, while in other cases a clerical review will be required to decide whether (and how) the data should be reordered. Testing using the 2019 census rehearsal suggests that this process may correct around 1000 errors (out of around 2 million households), which may otherwise cause problems later in processing and also affect the quality of the data (problems with the relationship matrix is the main issue discussed on the 2011 Data Quality Issues web page: https://www.scotlandscensus.gov.uk/data-quality-issues).

3. Introduction and Background

In 2011 for the paper returns name was captured on the household form and the individual person forms. Respondents also were invited to enter the relevant name on the relationship matrix, although these were not captured. Respondents were requested to enter people in each of these areas in the same order. It was therefore assumed that person one on the relationship matrix is the same as person one on the individual forms, and so on. For the 2019 census rehearsal, the guidance was improved to make it clearer that respondents should enter people in the same order throughout the form.

However this may not be the case: respondents may enter people in different orders in different part of the form. This can lead to relationship matrix information being attached to the wrong person. In some cases this might be detected if it leads to an implausible relationship. However this puts extra burden on that process, especially as that would likely require manual review of the scanned form. In addition, incorrect relationships could end up being used in outputs (if they are not obviously incorrect).

Correcting these problems in ordering would therefore have the following benefits:

1. improved data quality by minimising cases where individuals are connected to the incorrect person on the household form and hence having incorrect relationship matrix information, and
2. avoiding the need for clerical review of implausible relationships, saving time and resource downstream in data processing.

In 2022 it is expected that most census returns will be online. Online collection will ask respondents to enter names once, and then these will be used throughout the form. This will help avoid the above problems (and any that do remain will be undetectable). However it is expected that there will still remain many paper forms and these could have the same problems as in 2011.

Another difference from 2011 is that the paper forms will capture the respondent's name on the household form, individual form *and* on the relationship matrix. This will allow for a direct comparison between the individuals on the relationship matrix and the individual forms. This will allow us to detect and correct any differences in ordering, ensuring that the relationship matrix information is attached to the correct persons. The test on 2011 data therefore only includes comparisons between the individual form and household forms, while the test on the 2019 rehearsal data also includes comparisons between the individual forms and relationship matrix.

4. 2011 Method

In 2011 there was no equivalent step. Some cases where the persons appeared in different orders were resolved manually, when incorrect ordering lead to conflicts in the data. This could occur if a person did not appear to be younger than their parent. However, this was a time consuming process, and could only detect cases that were obviously incorrect.

5. Proposed 2022 Method

5.1 Method Summary

In summary, the method proposed is to measure the similarity for each name on the individual forms with each name on the relationship matrix and with those at the start of the household forms. (As relationship matrix names were not captured in 2011 the testing on the 2011 data involves linking the names from the individual forms with those at the start of the household forms.) These similarities are then compared to find the optimal ordering of household names to link to individual person names. If there are multiple optimal orderings then we do not simply want to select one at

random. Thus in such situations the algorithm would send the case for clerical review.

If an optimal ordering is found that is not the default ordering (i.e. household person 1 assigned to person 1 on the individual forms, person 2 to 2, and so on) then that ordering is suggested. In many cases this ordering could be accepted without review. However the following situations would probably indicate that review is required:

- The total cost of the suggested ordering is not a substantial improvement on that of the default ordering
- There is no unique optimal ordering (generally when there are multiple names the same)
- The optimal ordering involves a link with a high cost, perhaps suggesting that something has gone awry.

5.2 Method Detail

This subsection presents the algorithm used to find the optimal solutions. A worked example of this algorithm is given below to aid explanation. In addition, a diagram summarizing the algorithm is given in Figure 1. In addition, pseudocode for the main algorithm (not including the linking algorithm) is included in Annex 2: Pseudocode of Main Algorithm. In the text below, references are made to the relevant line numbers in this annex.

The algorithm loops round all the available households (line 1). Single-person households are not considered as these cannot be reordered. Households with more than 5 persons will be returned using a main form and one or more continuation forms and so are considered separately. We plan to consider each continuation form as a separate household, and consider the names within each continuation form for reordering. The results below do not include these households. Households where a different number of people appear in the different parts of the form are also problematic, so are also not considered. Some of these may need resolved during the Resolve Multiple Returns step.

For each household all the names from the individual forms and all from the household from are loaded into arrays. Each name from the individual forms are compared with each name from the household form. Their similarity is measured using the name part of the linking method, which is shared with other similar linking tasks, and is described in Annex 1. For each name component (first, middle and last name) this calculates a score (line 7) indicating the strength of evidence for the pair being a match (hereafter the *for score*), and also a score indicating the strength of the evidence against a match (hereafter the *against score*). The for scores and the against scores are each non-negative, and for each component only one of them can be greater than zero. Exact agreement on first or last name gives for scores of 50 (and against scores of zero) while exact agreement on middle name gives a for score of 25. If the name component is missing on one or both of the names then the for and against scores would both be zero. (An exception is for middle names. If Exactly one of the name is missing then the scores are zero. However if both are missing then it may be that the person does not have a middle name. Such cases get a for score of half that of exact agreement (i.e. 12.5).)

These scores are combined into a single variable (hereafter referred to as the cost, lines 8–9). The costs are then stored in a array. The costs are calculated as:

$$c = 125 - (f_f + f_m + f_l) + 2(a_f + a_m + a_l)$$

where c is the cost, f_f, f_m and f_l are the for scores of the first, middle and last name components respectively, and a_f, a_m and a_l are the against scores of the first, middle and last name components respectively.

The cost function was chosen so that the scores for the first, middle and last names contributed equally (recall that the score for the middle name has a smaller range, with the maximum for score being 25 instead of 50). The against scores have positive coefficients, while the for scores have negative coefficients. This means that larger cost values indicate greater differences between the names (and stronger evidence of a non-match). The against scores are given double the weighting of the for scores. Thus if any of the name components are different enough that it is more

likely that they are different names then that would influence our decision on whether these were the same more than how similar the name components that were similar are. That is, if any of the name components are clearly different then it would not much matter if the other components were the same, as they are still likely to be different persons. Finally, the constant of 125 is added so that the minimum possible cost will be 0 (in the case where the names agree perfectly).

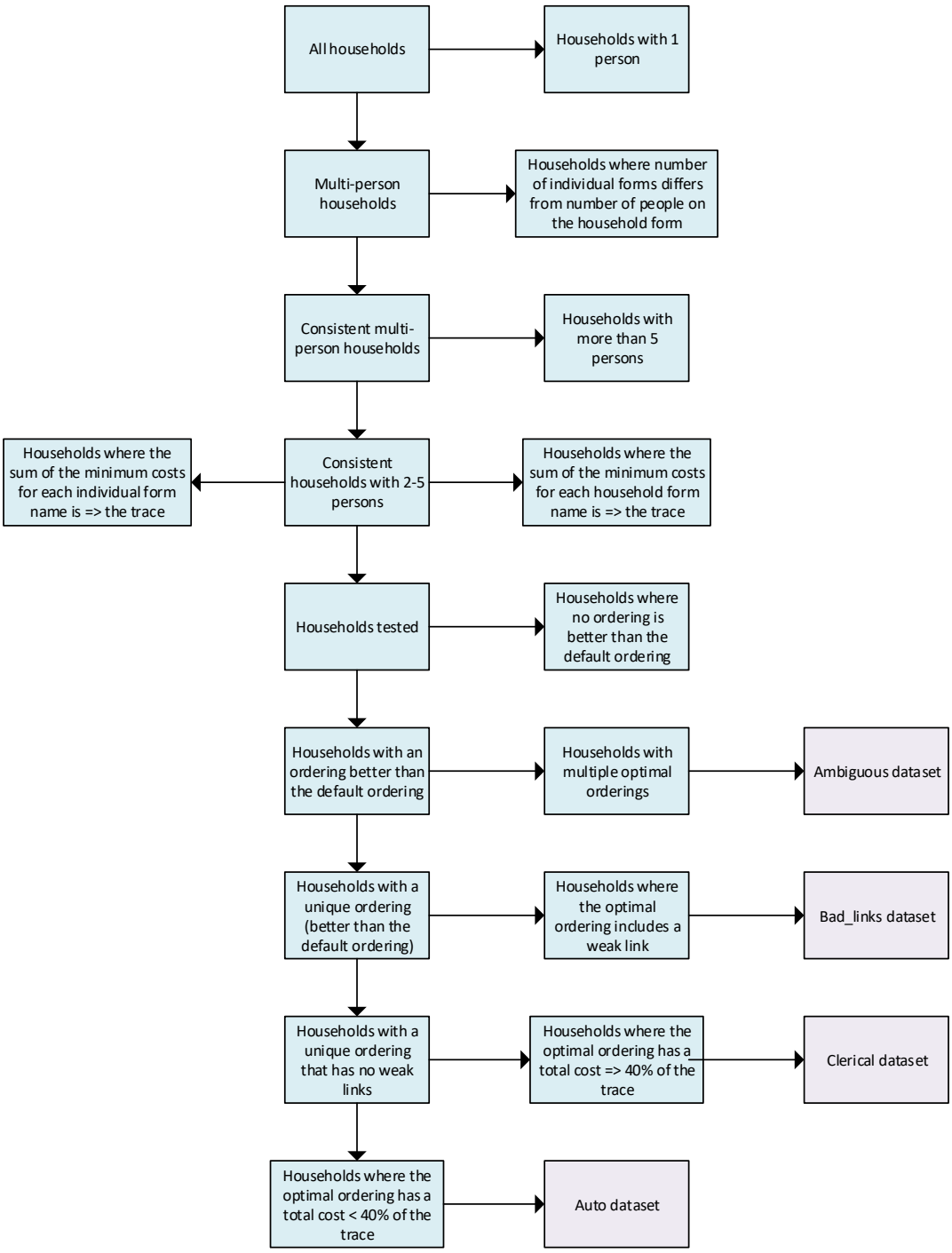

Figure 1 Summary of algorithm. Downward arrows indicate cases being passed to the next stage. Sideways arrows indicate cases being removed from the process.

Note that a cost of 0 is only possible when each of the names compared have a first, middle and last component. This means that John Smith to John Smith (cost of 12.5), would be considered weaker than John Robert Smith to John Robert Smith (cost of 0). This is what we want, because if we just have John Smith then they may have neglected to enter their middle name. Similarly we would want John Smith to John Smith (cost of 12.5) to be stronger than John Robert Smith to John Smith (cost of 25). This is because in the first case it may simply be someone without a middle name, while in the second case we know that there is someone with a middle name, and these are consistent only if they neglected to enter their middle name.

What we are seeking is a set of links connecting all the names on the household form with all the names on the individual form, the total cost of which is smaller than that of any other possible set of links. Each set of links can be thought of as a reordering of, say, the household names, and so is referred to as an ordering.

The first test is around the plausibility of the optimal ordering not being the default ordering (i.e. where person 1 on the individual form linked with person 1 on the household form, and so on). The total cost of the default ordering is the sum of the cost of person 1 to person 1, person 2 to 2, and so on. Thus, if we lay out all the costs in a matrix, then the total cost of the default ordering would be the trace[1] (calculated at line 10). To do this we consider the strongest link (i.e. that with the lowest cost) for each of the *n* names on the individual forms, where *n* is the number of individuals in the household (lines 11 and 13). If the total of these minimum costs is greater than or equal to the trace then there will not be an optimal ordering that is better than the default ordering, so there is no point searching for one. A similar test can be done by considering the household names (lines 16–23). If neither of these tests rule out the possibility of a better ordering then we proceed to the search (line 25).

Recall, *n* is the number of individuals in the household (HH). For each of the *n* names on the individual forms, there would be *n* possible names on the household form. (Forms where the number of people on the household form differs from the

[1] See https://en.wikipedia.org/wiki/Trace_(linear_algebra).

number of people on the individual forms are excluded from the analysis. These cases may be considered again after the Resolve Multiple Returns cleansing step, which may bring these numbers into alignment.) The search algorithm loops through the n^n possible orderings (line 28). For each possible ordering it goes through each of the n names from the individual forms (line 35). Not all of these are valid orderings[2], as some would use the same name from the household form for different names from the individual forms. The algorithm identifies the corresponding record from the household form (line 37). If that HH-form record has already been used in the ordering then it is considered invalid and we can proceed to another ordering (see lines 38–43).

If the individual record at which the ordering invalidity became apparent is not the last one then we can skip some orderings (purely to improve efficiency, see line 58). For example if we get to 2, 2, 1 (i.e. name 1 on the individual form linked to name 2 on the household form, individual name 2 linked to household name 2, and individual name 3 linked to household name 1, see following subsection for an example) then we can see that this is not a valid ordering as the second name from the household form appears twice. There is no point in moving to the next ordering (2, 2, 2) or indeed the one after (2, 2, 3) as these both have the same problem. Therefore we can move on to ordering 2, 3, 1.

As we loop through the individual records we keep a running total of the costs (see above) of their links to the HH-form records (line 39). If, at any point, this running total exceeds the total cost of the best known ordering then the total cost for the full ordering will be greater than the current best total cost (as the costs cannot be less than 0) and so this cannot be an optimal solution. Therefore, we can stop and move to another ordering in the same way as for invalid orderings (see line 35).

If we get to the end of the ordering (line 45) and still have a total cost that is better than the current best total cost (line 46) then we update the current best total cost (line 48) and make a note of the ordering (lines 50–53). We also indicate that this

[2] Actually there are $n!$ valid orderings. However it is simpler to consider the n^n options and ignore the invalid orderings.

ordering is a unique optimal solution (line 47). If the total cost only equalled the current best total cost (line 54) then we indicate that there are multiple optimal orderings (line 55), i.e. there is not a unique optimal ordering.

Once all the possible orderings have been considered (or skipped) then we compare the current best total cost with the trace (line 60) and output to one of four datasets (lines 61–62) if the current best total cost is less than the trace (i.e. if the default is not an optimal ordering). If the optimal ordering is unique then we do two more checks. The first is to check that none of the links are particularly bad (that is, links with a cost above 150), as this would raise suspicion that something was awry with the solution, so these go to a dataset called *bad_links*. Next there is a check to see how much of an improvement the optimal ordering is over the default ordering. Only if it has a total cost of less than 40 per cent of the trace is it sent to the *auto* dataset for automatic acceptance, with the remaining going to a *clerical* dataset for clerical review. (The 40 per cent threshold was set following manual examination of cases. It was found that all cases below this threshold should be reordered, but some cases above this should be left as the respondents had indicated.) Finally, if there are multiple optimal orderings then it is saved to an *ambiguous* dataset, which should also be clerically reviewed.

Following clerical review the data can be fed back to the census data (in the Core Data Store). This could be done by creating a new relationship matrix variable and setting the value of this depending on which name the individual is linked to. Households where the clerical reviewer considered that the original order was the correct order would have their new relationship matrix variable set to the same as the original relationship matrix variable.

5.3 Worked Example

Table 1 gives a fictitious example of a form where the people appear on the household form in a different order from on the individual forms. These therefore need reordered.

Step 1:

This household has the same number of names on the household and individual forms and this number (three) is between two and five. Therefore this household is eligible for consideration and we proceed to Step 2.

Table 1 Example dataset of names from the household and individual forms.

Household form		Individual forms	
Person number	Name	Person number	Name
1	John Smith	1	Jane Smith
2	Jane Smith	2	John Smith
3	Mary Smith	3	Mary Smith

Step 2:

Each name from the household form is compared with each name on the individual form. During this process the "for" and "against" scores for the first, middle and last names are calculated and these are then combined into a single variable called cost (see Table 2).

Table 2 Costs of the links between the names on the individual forms and the names on the household form.

Name on household form	Name on individual form			Minimum cost
	Jane Smith	John Smith	Mary Smith	
John Smith	90.6	12.5	96.3	12.5
Jane Smith	12.5	95.9	69.2	12.5
Mary Smith	69.2	96.3	12.5	12.5
Minimum cost score	12.5	12.5	12.5	

Notes to table:

Grey cells indicate the combinations that constitute the default ordering. The total of these costs is the trace. The marginals show the minimum cost for each household-form name and individual form name. The sum of the values in the blue cells is the total minimum costs for the household-form names. The sum of the values in the green cells is the total minimum costs for the individual-form names.

Proceed to Step 3 (using the costs in Table 2).

Step 3:

Calculate the trace (grey cells in Table 2)

This is the total cost of the default ordering (when we link person 1 form the household form to person 1 on the individual form, person 2 to person 2, and so on).

Trace = 90.6 + 95.9 + 12.5 = 199.3.

Step 4:

Calculate the minimum cost for names on the households form (blue cells in Table 2)

Total Minimum Cost for Households: 12.5 + 12.5 + 12.5 = 37.5.

Step 5:

Calculate the minimum cost for names on the individual forms (green cells in Table 2):

Total Minimum Cost for Individuals : 12.5 + 12.5 + 12.5 = 37.5.

Step 6:

Compare total minimum costs to the trace

The total minimum cost for household-form names (37.5) and the total minimum cost for individual-form names (37.5) are both less than the trace (199.3). Therefore we proceed to Step 7.

Step 7:

Search for optimal solutions

Set current best total cost to the value of the trace (199.3).

Loop round orderings (theoretically $3^3 = 27$, although many of these can be skipped). See Table 3.

Table 3 Indication of how the algorithm would loop round orderings in the example (see Table 1 and Table 2).

Ordering	Result
1, 1, 1	Invalid ordering (HH name 1 (John Smith) used for the second name as well as the first). Therefore: • increment the second link to get to 1, 2, 1
1, 2, 1	Invalid ordering (HH name 1 (John Smith) used for the third name as well as the first). Therefore: • increment the third link to get to 1, 2, 2
1, 2, 2	Invalid ordering (HH name 2 (Jane Smith) used for the third name as well as the second). Therefore: • increment the third link to get to 1, 2, 3
1, 2, 3	Total cost = 199.3. This equals the best total cost so: • indicate that the optimal ordering is not unique • increment to ordering (1, 3, 1)
⋮	
2, 1, 3	Total cost = 37.5. This is smaller than the best total cost (199.3). Therefore: • amend best total cost to 37.5 • record 2, 1, 3 as the optimal ordering • indicate that the optimal ordering is unique • record the cost of the link(s) with the highest cost (12.5) • increment to ordering (2, 2, 1)
⋮	
2, 3, 1	The costs of the first two links (1–2 and 2–3) are 12.5 and 96.3 so the running total (108.8) is already above the best total cost (37.5). Therefore: • increment the second link; as this takes us beyond the possible range then increment the first link and reset others (i.e. to 3, 1, 1)
3, 1, 1	The costs of the link (1–3) is 69.2 so the running total is already above the best total cost. Therefore: • increment the first link; as this takes us beyond the possible range we are now done

The current best total cost was last updated for combination 2, 1, 3, where the total cost was 12.5 + 12.5 + 12.5 = 37.5. This was marked as a unique optimal ordering. We also kept a note of the cost of the link(s) with the highest cost in this ordering (in

this case 12.5). No other orderings were found that are as good as this, so this remains the unique optimal solution.

Step 8:

Compare current best total cost with the trace
Once the combinations have been considered the current best total cost (37.5) is compared with the trace (199.3). As it is better (lower) than the trace then this we proceed to Step 9.

Step 9:

Save the household with the optimal ordering. In this case, the order is now 2,1 3.

As the solution is marked as unique in this example, the best total cost score is better than 40 per cent of the trace (40% of 199.3 = 79.72), and there are no bad links then it is considered to not require clerical review and so is output to *auto*.

6. Results Using 2011 Data

This algorithm was run on step 6 data for processing unit 2 (PU2). PU2 covers East, North and South Ayrshire. This PU was chosen as data was available for the records that were deleted by the Resolve Multiple Returns process in 2011 (during step 6), along with those that were retained.

Table 4 Number of households and records identified by the algorithm, broken down by the classification. Also shown is the number of cases where the ordering should be changed.

Dataset	Number of households	Number of changes	Number of records
Auto	1035	1035	3162
Ambiguous	39	31	156
Clerical	87	61	286
Bad links	28	14	86
Total	1189	1141	3690

The number of households directed to each of the different datasets is given in Table 4. 87 per cent of the 1,189 considered households (those with 2–5 people) are considered to have sufficient evidence for the new ordering to be accepted without review.

A selection of a few hundred cases from the auto dataset were reviewed. In all these cases the suggested ordering appeared to be the correct one. For each of the other datasets all of the cases were reviewed.
In the end 1,141 households had changes. As there are 10, roughly equally sized processing units, across the whole census this would be around 11,410 changes. The 2011 census was primarily a paper return census.

7. Results Using Rehearsal Data

In the rehearsal data, as in the 2022 data, the names are captured in three locations: household form, the relationship matrix and on the individual forms. The code was therefore modified to run through the algorithm twice for each household, once comparing the names on the individual forms to those on the household form, and secondly comparing the names on the individual forms to those on the relationship matrix.

Table 5 Breakdown of households by whether household names or relationship matrix names needed reordered, and what category they were. Cells shaded green are those where some part could be automatically reordered.

Individual–Household	Individual–Relationship Matrix					Total
	Auto	Ambiguous	Bad link	Clerical	None	
Ambiguous		168		2	91	261
Auto					1	1
Bad link				1	1	2
Clerical	1	7		18	24	50
None	1	492	8	76	13,446	14,023
Total	2	667	8	97	13,563	14,337

The rehearsal data contains around 38,000 individual records covering 22,838 households. However, after removing single-person households, households with more than 5 people, and households where the number of people is different at the different parts of the form, this reduces to 14,337. Table 5 below shows these households broken down by whether the household names or relationship matrix

names needed reordered to match the names on the individual forms, and if so whether they needed reviewed, and if so why. It can be seen that in the vast majority of cases (13,446, 94 per cent) no reordering is suggested. It can also be seen that the majority of the remaining cases (492 + 168 + 91 = 751, 5 per cent) are where it is being flagged because it is ambiguous, that is, when there are multiple identical names. These are generally cases where the names are blank due to issues with the paper scanning. (The scanning sometimes detects the boxes printed on the form to indicate where respondents should enter letters. This results in blank fields being recorded with characters, typically I or l.) Hopefully this issue will be corrected in 2022 but if not most of these cases will likely be removed when false returns are removed. Then the name reordering task could then be run following that step. This should substantially reduce the amount of clerical review required.

The remaining cases were all clerically reviewed. In the three cases flagged for automatic reordering (shaded green) it was found that the suggested order appeared correct. Also, five cases from among those passed for clerical review appeared to need reordering. The majority of the remainder appeared to be due to issues with paper scanning. It is difficult to know at this stage how much review would be required if the process was run on data where the scanning problem had been resolved.

The rehearsal included 38,000 individual records. If the population of Scotland is around five million then to scale up to the full population we would need to increase our findings by around 130. Scaling the eight cases up to the full census would therefore suggest that or order 1000 households could have their ordering corrected using this method. This is lower than suggested by the 2011 testing. This may be because a higher proportion of 2011 cases were paper returns, and/or because the guidance around entering people in the same order was clearer in the rehearsal than it had been in 2011.

8. Strengths and Limitations

The method presented here can correct perhaps about 1000 problems (out of around 2 million census households) when attaching the relationship information to the individual census returns. This would be done before incorrectly attached information causes problems later in processing. As such it can save time performing manual investigations and changes, and potentially unpicking some of the processing that has already happened. Many of these changes can happen automatically, without the need for human involvement, and this process appears to be robust. Despite not being fully optimized the linking processes is projected to run on the full census in about 20 minutes.

The process does, however, require some clerical review, the amount of which is very difficult to estimate with the available data (the 2011 data has a different pattern of paper returns and the rehearsal data had problems with paper scanning). It may turn out that some of this clerical review is not needed if it would not affect the relationship matrix (for example in households with just a married couple). If this was considered to be an issue then the clerical review could be cut down to only those households where reordering would affect the relationship matrix.

A further check is currently being explored. This will make use of the actual relationships that appear on the relationship matrix, and the ages derived from the dates of birth that appear on the individual forms. Once an optimal ordering has been identified the relationships and ages will be considered. If there are any conflicts (or implausible comparisons) then the case will be passed for review.

9. Conclusion

For a modest outlay this process can correct potentially many problems in attaching information from the relationship matrix to individual records. This can be done at an early stage in processing, before it causes problems in later stages. It may be that it needs to be done following the stage where the blank records from paper scanning get removed. In general, though, it is useful for this step to happen at the start of

processing, before other cleansing tasks, as it can help avoid problems at those steps.

Many of the identified cases can be reordered automatically without human input. The remainder would be clerically reviewed, perhaps only if the ordering would affect the relationship matrix. We have therefore recommended that this process be carried out in the census 2022 processing.

Annex 1: Scoring of Name Comparisons

This section discusses in detail how the for scores (which indicate the strength of evidence for two records representing the same person) and the against scores (which indicate the strength of evidence for two records representing the different persons) are calculated for the various components. There are a number of attempts to find evidence for a match. Each one will update the for and against scores only if that will strengthen the evidence for a match.

Missing Names

If name is missing on one or both records then the "for" and "against" scores are both 0. Otherwise if a name component is exactly the same between the two records then the for score is 50 (25 for middle name) and the against score is 0.

For first names there is also a check for the name being "BABY" on both records. In this case the "for" and "against" scores are both set to 0 as the guidance (in 2011) indicated that unnamed infants should be recorded as "BABY". This scoring was developed for the Resolve Multiple Returns step, where two BABYs could easily be twins. In the Census Coverage Survey linking the likelihood may be different so this may need revised.

Nicknames

Another check for first names is nicknames. Thus if we had "Alexander" on one record and "Sandy" on the other then it is quite plausible that these are the same person, even though the first name strings are quite different. To perform this check we make use of the nickname linking variable. That variable is set to a particular value for a range of names that have the same nickname. Thus if first was either "Alexander" or "Sandy" (or "Alex", "Xander", and others) then the nickname variable is set to "Alexander". (The name groupings were built up manually, assisted by exploring links between datasets where last name, date of birth and postcode agreed, but first name did not.) Thus if the first names differ between records but the nicknames agree then the against score is set to 0 and the for score is set to 20.

Some of these are specific to a particular sex. Thus if the first name is "Alex" then the nickname will be set to "Alexander" if sex is male and "Alexandra" if sex if female. There is also a second nickname variable that groups together more tenuous name groupings such as "John" and "Ian", which results in a for score of 10.

The nickname check also detects alternate spellings of the same name, such as "Nicholas" and "Nicolas". This may be particularly important for Census Coverage Survey linking when data is reported verbally and spellings may not be confirmed. In total there are 189 groupings defined, and 45 more tenuous ones.

Character comparison for names

If none of these situations hold then the name components in the two records are compared at the character level using a method inspired by the Damerau–Levenshtein edit distance[3]. The characters in the name from one record are linked to those in the name from the other record. This is done by first comparing the characters at the same location in the strings. If these do not agree then this moves to adjacent letters, and then letters at a distance of two, and so on. Once this has completed there is a tidying up stage to ensure that adjacent letters are linked to letters at the same distance if possible.

Once the letters have been linked they are then analysed in order to identify the substitutions, transpositions, deletions, insertions and jumps would be required to transform one string into another. For each of these there is an associated score. These scores depend on the letters involved. For example if we need to insert a "W" then that would attract a larger penalty than if we only need to insert a "I" because a mark on a page may be mistaken for an "I" in scanning, but is unlikely to be mistaken for a "W". Similarly for substitutions some changes are more plausible than others. Combinations like "U" and "V" can be easily confused, as can "O" and "D". In total 50 such combinations are noted.

[3] See https://en.wikipedia.org/wiki/Damerau%E2%80%93Levenshtein_distance for a general discussion.

The scores from all the individual differences are then combined to give an overall score. That score is then converted to scores for and against the records being a match.

Swapped first and last names

Sometimes people enter their names in an unexpected order. To account for this a comparison is made between the first name of one record and the last name on the other record and vice versa. If these both agree then the for scores for both first and last names are set to 40. If only one of these agrees then one of these scores is set to 40, while the other is set by doing the character comparison on the differing values. That is, if first_1 agrees with last_2 then the first for score will be 40, while the last for score will be set by doing a character comparison between first_2 and last_1.

Titles

If first name begins "MR " or "MRS " then that part is removed from the first name and stored in a variable called title. If the two records being compared both have "MR" and "MRS" respectively in their title variables, and their sex agrees with this information, then a penalty of 20 is combined with the for and against scores for first name.

Comparison to middle name

Some people go by what is officially their middle name. In order to successfully link these cases the first name for one record is compared with the middle name of the other. If this agrees then the for score for first name is set to 15 (unless it was already over 15). A similar check is also done between last name and middle name.

Compare name parts

Some people have double-barrelled first or last names. However they may go by only part of this. For example "Sarah-Jane" may go by Sarah, or even Jane. To detect such cases we make use of other linking variables that pull out parts of names

that are delimited by special characters. If these agree with the name from the other record then the for score is set to 25 (unless it was already over 25). This is done for first names and also for last names. In other comparisons special characters (including spaces) are removed before the comparison is made.

Comparing first letters of name or Double Metaphone code

The next check is to count the number of letters that agree at the start of the name from the two records. If so then the for score is set to be that given in Table 6. This covers a range from one letter agreeing to five (or more) letters agreeing. If only one letter agrees then this is treated differently, so that this method is used only if one record only has the initial (e.g. if one record had "Peter" and the other had "P", but not if the other was "Paul"). These scores are only used if they result in a higher for score than would otherwise be. Another exception is when 3 or fewer letters agree and the names are distinct but common. For example if we had Mary and Margaret then the first three letters agree, but as the names are common then this is not used to score the similarity.

Table 6 The for scores assigned when the first part of the name agrees either on the name itself, or the Double Metaphone coding of it. If only one letter agrees then this method is only used if one of the records only has one letter.

Number of characters agreeing	Name	Double Metaphone of name
5+	20	20
4	13	13
3	7	9
2	3	4
1*	10	-

Similarly the first characters of the Double Metaphone are compared. The Double Metaphone is a phonetic code[4], so this allows for detection of cases where a name has been written differently, but sounds the same. This is another situation that may be particularly common for verbally reported data such as the Census Coverage Survey. As a character in the Double Metaphone code can relate to more than one letter in the original string, agreement on Double Metaphone can indicate stronger agreement than agreement with the same number of letters on the original string.

[4] See https://en.wikipedia.org/wiki/Metaphone#Double_Metaphone for a general discussion.

Therefore these scores are slightly larger than the equivalents for the agreeing letters on the original name.

There is an exception when comparing the last names on the original string or Double Metaphone. If the last name begins “Mc” or “Mac” then the count of the agreeing characters is reduced by 2 and 3 respectively. This is because names beginning this way are so common, while being very distinct. Therefore we would not want to say that MacDonald and MacPherson were as similar as Scalon and Scanlan.

Full name

Sometimes a space is missing between the first and middle name, meaning that the middle name gets concatenated onto the first name. Other times a space gets inserted between letters of the first name, meaning that part of the first name gets put as the middle name. Another issue is that the whole name can be entered in the first name field.

All these issues can be resolved by considering the full name, i.e. the concatenation of first, middle and last names (with spaces and other special characters removed). This full name is one of the linking variables used. It is compared between the two records. If it is not exactly the same then a character comparison is done. This allows a for and against score to be calculated for the full name. If this score is better than the for scores for first and last name then the first and last for scores are amended using the full name for score.

Annex 2: Pseudocode of Main Algorithm

Bold text summarizes further detail that is not described here.

```
1   Do h=1,number_of_households
2    Trace=0
3    Total_best_cost_i=0
4    Do i=1,n
5     Best_cost=1000
6     Do j=1,n
7      Measure similarity between i on individual form and j on HH form
8      cost(i,j)=125-(first_for+middle_for+last_for)
9                  +2*(first_against+middle_against+last_against)
10     If i==j then trace=trace+cost(i,j)
11     Best_cost=min(Best_cost,cost(i,j))
12    Enddo
13    Total_best_cost_i=Total_best_cost_i+Best_cost
14   Enddo
15
16   Total_best_cost_j=0
17   Do j=1,n
18    Best_cost=1000
19    Do i=1,n
20     Best_cost=min(Best_cost,cost(i,j))
21    Enddo
22    Total_best_cost_j=Total_best_cost_j+Best_cost
23   Enddo
24
25   If Total_best_cost_i < trace and Total_best_cost_j < trace then
26    n_comb=0
27    Best_tot_cost_so_far=trace
28    Do until (n_comb ≥ n*n)
29     Do i=1,n
30      Used(i)=0
31     Enddo
32     Tot_cost=0
33     i=0
34     Finish=0
```

```
35     Do while (i < n and tot_cost ≤ best_tot_cost_so_far and finish=0)
36      i=i+1
37      j=1+mod(floor(n_comb/(n**(n-i)),n)
38      If used(j)==0 then
39       Tot_cost=Tot_cost+cost(i,j)
40       Used(j)=1
41      Else
42       Finish=1
43      Endif
44     Enddo
45     If i==n and finish==0 then
46      If tot_cost < best_tot_cost_so_far then
47       Unique=1
48       Best_tot_cost_so_far=tot_cost
49       Worst_link=0
50       Do q=1,n
51        Linked(q)=1+mod(floor(n_comb/n**(n-q)),n)
52        Worst_link=max(worst_link,cost(q,linked(q)))
53       Enddo
54      Else if tot_cost==best_tot_cost_so_far then
55       Unique_comb=0
56      Endif
57     Endif
58     N_comb=n_comb+n**(n-i)
59    Enddo
60    If best_tot_cost_so_far < trace then
61     Output to one of the files depending on unique_comb, worst_link,
62      and best_tot_cost_so_far/trace
63    Endif
64   Endif
65  Enddo
```

Annex 3: Glossary

Term	Definition
Ordering	A way of assigning names from the household form to the names from the individual forms. These are represented as a vector with each element indicating which household-form name is allocated to each individual-form name. Thus (3, 1, 2) indicates that: • the first person from the individual form has the third name from the household form assigned (linked) to it • the second person from the individual form has the first name from the household form assigned (linked) to it • the third person from the individual form has the second name from the household form assigned (linked) to it.
Default ordering	The ordering where the *i*th person from the individual form is linked to the *i*th person from the household form: (1, 2, 3, … n)
Valid ordering	An ordering where each name from the household form appears exactly once.
Similarity	A quantified measure of the level of agreement between name components for different names (e.g. first name for one record and the first name of another record). Calculated separately for first name, middle name and last name.
Cost	A single measure of the similarity of two full names. This is calculated from the similarity scores for first, middle and last name parts.
Total cost	The total of the costs of all the links in an ordering.
Trace	The total of the costs of all the links in the default ordering.
Strong link	A link where the linked names are particularly similar, i.e. there is a low cost.
Weak link	A link where the linked names are dissimilar, i.e. there is a low cost. If the cost is greater than 150 then the link is classed as a bad link.
Current best total cost	The total cost of the best ordering out of the orderings thus far considered for the household (including the default ordering).

Optimal ordering	An ordering with the lowest total cost.