Catalogue no. 11-522-X
ISSN 1709-8211

Proceedings of Statistics Canada Symposium 2022:
Data Disaggregation: building a more representative data portrait of society

A case study of using Splink: Census duplicate matching

by Mary Cleaton, Johanna Hall, Rachel Shipsey, Zoe White and Kristina Xhaferaj

Release date: March 25, 2024

Statistics Canada Statistique Canada

Canada

Proceedings of Statistics Canada Symposium 2022
Data Disaggregation: Building a more representative data portrait of society

A case study of using Splink: Census duplicate matching

Mary Cleaton, Johanna Hall, Rachel Shipsey, Zoe White, Kristina Xhaferaj[1]

Abstract

The authors used the Splink probabilistic linkage package developed by the UK Ministry of Justice, to link census data from England and Wales to itself to find duplicate census responses. A large gold standard of confirmed census duplicates was available meaning that the results of the Splink implementation could be quality assured. This paper describes the implementation and features of Splink, gives details of the settings and parameters that we used to tune Splink for our particular project, and gives the results that we obtained.

Key words: Splink, census, probabilistic linkage, Fellegi-Sunter, Expectation-Maximisation

1. Introduction

1.1 Introduction to Splink

Splink is an open-source Python package that was designed in 2020 by the UK Ministry of Justice (MoJ) to facilitate large-scale data linkage projects. It is publicly available via GitHub (Linacre, et al., 2022) and is compatible with AWS Athena and Spark, enabling linkage of Big Data. Splink implements the Expectation-Maximisation (EM) algorithm (Dempster, et al., 1977) and the Fellegi-Sunter method of probabilistic data linkage (Fellegi & Sunter, 1969).

1.2 Census duplicates

In the 2021 England and Wales Census (henceforth '2021 Census'), it was estimated that there were approximately 420,000 duplicate records, contributing to an overall overcount of 0.96% (ONS, 2022). This linkage project concentrates on locating duplicate records at different geographical locations. This type of duplication can occur in a census for a variety of reasons including when people:

- Move to a new house during the census period and complete a census return for both addresses.
- Have multiple residences and complete a census return for at least two of them.
- Are students with a term-time and a home address and complete a census return for both of these.
- Are children who spend some time living with each of their separated parents.

1.3 The gold standard

Following the 2021 Census, ONS used a simple scoring strategy based on date of birth, first name and last name to identify potential duplicates in samples of the census data. These candidate duplicates were further refined using a novel classification algorithm called the Automatic Checking Algorithm (ACA) (Shipsey & White, 2020). The ACA split the candidate duplicates into 'automatically accept', 'automatically reject' or 'uncertain' categories. Those candidates classified as uncertain were reviewed clerically. This exercise produced a set of 3,219,099 sampled census records that were either classified as a duplicate (23,434 records) or a non-duplicate (3,195,665 records). Records and their duplicates (where applicable) were collated to create a gold standard dataset.

[1] All authors: Methodology and Quality Directorate, Office for National Statistics, England and Wales, UK, NP10 8XG (datalinkage@ons.gov.uk).

1.4 Goals and aims

This project's goals were twofold. Firstly, to test an alternative method of identifying duplicates in a dataset, using the 2021 Census as a case study. Secondly, to showcase Splink, thereby providing a case study of using Splink for real-world Big Data linkage. This paper describes our use of Splink but does not further mention the alternative overcount method.

2. Splink

Splink is an implementation of the Fellegi-Sunter algorithm (Fellegi & Sunter, 1969), which is the standard method of probabilistic data linkage used worldwide. We used Splink v2.1.4 and gave feedback to the development team at MoJ. Subsequent improvements have been made to the package which have since been released as Splink v3.

2.1 m- and u-values

The Fellegi-Sunter method is equivalent to a Naïve Bayes algorithm. It requires m- and u-values to be input for each variable used for matching. The m-value is the probability that the values for a given linkage variable will agree when the candidate pair is a true match. The m-values can either be calculated from training data, such as a gold standard dataset, or via the EM algorithm. Within Splink, the local models can be used to calculate m-values if no gold standard dataset is available.

The u-value is the probability that the values for a given linkage variable will agree when the candidate pair is a true non-match (i.e., agreement is purely by chance). The u-values can be easily calculated using the Splink function 'estimate_u_values'.

2.2 Case statements

SQL case statements are a series of if-else clauses that describe the agreement states for a given variable. For example, agreement states might be exact agreement, no agreement, or partial agreement as determined by a string comparator such Levenshtein edit distance or Jaro-Winkler similarity score. All SQL case statements used in this case study are shown in (Xhaferaj, 2022).

As an example, the case statement for the 'first name' variable uses the following logic:

- If either first name is missing, then agreement level is -1.
- Else, if first names agree exactly, then agreement level is 3.
- Else, if Jaro-Winkler score between first names is ≥ 0.88 or first and middle names are transposed, or first and last names are transposed, then agreement level is 2.
- Else, if standardised Levenshtein edit distance between first names is ≥ 0.401, then agreement level is 1.
- Else, agreement level is 0.

2.3 Agreement weights

Using the m- and u-values, an agreement weight is calculated for each agreement level of the case statement as $log_2(m/u)$. Where a case statement clause is associated with non-agreement, the agreement weight will be negative. Where a case statement clause indicates partial agreement, the agreement weight may be positive or negative, dependent on the weights of the other clauses in that case statement. Table 2.3-1 shows the m and u values and the associated agreement weights for the 'first name' variable case statement described in Section 2.2.

Table 2.3-1
m- and u-values and agreement weights for the five agreement levels of the 'first name' variable

Agreement level	m-value	u-value	Agreement weight
3	0.7798	0.001490522	9.0311
2	0.1393	0.001386703	6.6502
1	0.0780	0.024442757	1.6736
0	0.0030	0.972680018	-8.3524
-1	N/A	N/A	0

2.4 Blocking

In theory, probabilistic linkage models compare every potential combination of records, using a Cartesian join. The majority of these comparisons are redundant as the records have little in common. Blocking is a technique used to bring together candidate pairs that share characteristics, thus reducing the search space and the number of comparisons required. A series of blocking rules are used to select the most promising comparisons. For example, blocking rule 1 requires an exact match on 'first name' and 'last name', plus 'middle name' to match or be missing; blocking rule 5 requires an exact match on region, 'first name initials', 'last name initials', 'date of birth' and 'sex'. Blocking rules must be loose enough to capture all potentially matching comparisons whilst being tight enough to produce a computationally feasible number of comparisons. Blocking is used in both the global model and local models in Splink.

2.5 The Splink global model

The Splink global model is an implementation of the Fellegi-Sunter method. The user inputs a prior value, which is an estimate of the probability of a randomly selected pair of records being a matching pair. The global model takes this prior, along with the m- and u-values, and uses a series of user-defined SQL case statements to filter candidate pairs and award the appropriate weight for each variable to the candidate pair. The weights are summed to calculate a final match-score. Typically, a threshold is chosen by user and candidate pairs scoring above the threshold are accepted as matches (aka duplicates, if matching a dataset to itself). The Splink global model can be fully customised by the user depending on their requirements, using the features outlined in Section 3.

In the global model, it is generally best to block using numerous comparatively tight rules that allow for all the different kinds of error in the data. This ensures that all possible matches are captured whilst reducing the number of candidate pairs that have to be scored. All global model blocking rules used in this case study are shown in (Xhaferaj, 2022).

2.6 The Splink local model

The Splink local model is an implementation of the EM algorithm. A series of local models are used to generate m-values for each of the matching variables and for each clause of the case statements. For each local model, a subset of the matching variables is used for blocking and the remaining matching variables are awarded m-values. For example, in local model 1, 'first name' and 'last name' might be used for blocking while m-values are calculated for 'middle name', 'date of birth', 'postcode' and 'sex'. In local model 2, 'date of birth' and 'postcode' might be used for blocking while m-values are calculated for 'first name', 'last name' and 'sex'. Some of the variables used for matching will have more than one set of m-values calculated, e.g., 'sex' in the example given above. As a default, Splink uses the harmonic mean of the different m-values. However, the user may choose to use a specific set of m-values if this seems more appropriate (see example in Section 4.6).

For each local model, the user must input a prior value, which is the estimated proportion of matches amongst the blocked records in that model. It is highly important that the local model's prior is appropriate. Otherwise, it may iterate towards a local maximum rather than the global maximum, resulting in nonsensical m-values. Users should always sense-check the m-values output by the algorithm and change the prior and/or case statements until they consider the outputs to be satisfactory.

3. Features of Splink

3.1 Splink allows for multiple agreement states

The original Fellegi-Sunter method only allowed for binary agreement states (i.e., full agreement or full disagreement), although modifications to the method permit the use of partial agreement states (Winkler, 1990). In contrast, Splink allows multiple agreement states to be used as default, with partial agreement states integrated seamlessly into the linkage process via SQL case statements. Allowing multiple levels of agreement in this way enables tailoring and refinement of linkages without the need for additional coding.

Splink's SQL case statements are an especially useful way of integrating partial agreement states as they permit nuanced options such as multi-variable logical statements. For example, 'unique property reference number' can

be used only where it is an exact match, with agreement otherwise deriving from the similarity between 'postcode'. This sort of complex, multi-variable case statement enables the use of combinations of variables that would otherwise breach the conditional independence assumption of the Fellegi-Sunter method.

3.2 Splink allows for term-frequency adjustments

Common values within variables can be accounted for using term-frequency adjustments (Winkler, 2000), which Splink includes by default. This can be particularly useful for string variables with cardinality skew (e.g., common values such as 'Smith' in the 'last name' variable of a UK dataset). Down-weighting is used to account for their high frequency within the dataset(s).

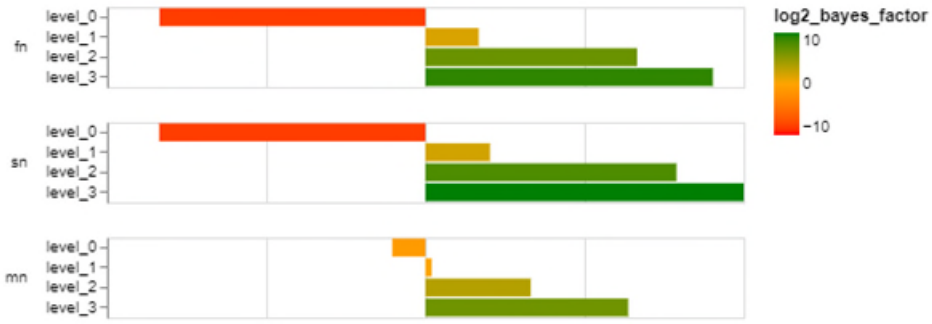
A new feature of Splink v3 is that the term-frequency adjustment can also be used to automatically adjust the agreement weights for categorical variables. For example, in England and Wales, agreement on the variable 'country of birth' does not increase confidence in the candidate pairs being a match if the country of birth is England or Wales. However, if 'country of birth' agrees and the value is Mexico, then this would increase confidence in the candidate pairs being a match. This feature was not available when we carried out our linkage; hence, we accounted for differences in cardinality by using different agreement levels in the case statements.

3.3 Splink produces a variety of outputs, including visualisations

Splink can produce a variety of outputs with which to explore the data, interrogate blocking decisions, sense-check results, view results summaries, tune results and provide greater transparency regarding how results were obtained. Two examples are given below. Further examples and explanations are available via the Splink GitHub site (Linacre, et al., 2022).

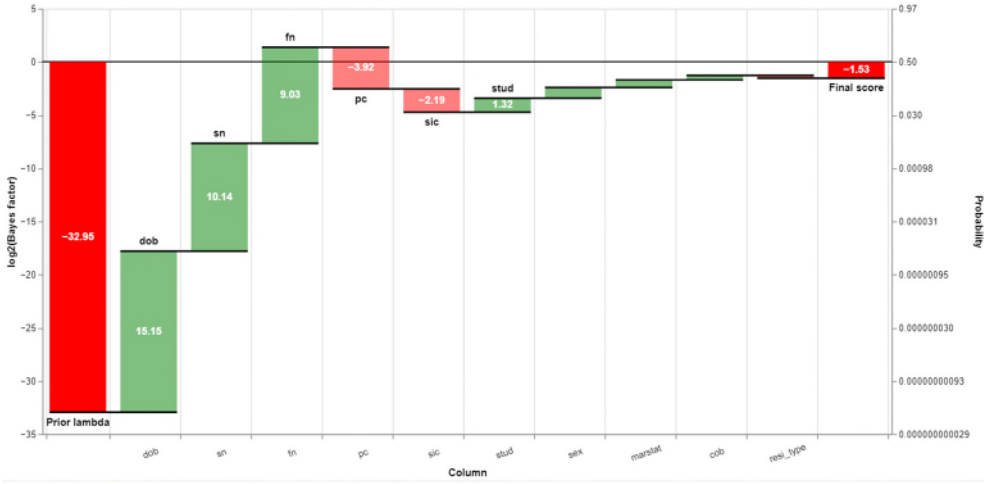
Example 1: Figure 3.3-1 shows the agreement weights for three different variables. It demonstrates that disagreement (level 0) on middle name is given a much less negative weight than disagreement on first or last name. This is as expected, as middle name is typically less well recorded in datasets than first or last name. This shows how this type of Splink chart can be used to sense-check agreement weights in comparison with each other. A full chart showing agreement weights for all of the variables in the case study is shown at (Xhaferaj, 2022).

Figure 3.3-1
Agreement weights assigned for the agreement levels of three variables: first name (fn), last name (sn) and middle name (mn)



Example 2: Figure 3.3-2 provides an example of a Splink waterfall charts for a chosen candidate pair. The prior is first applied (prior lambda). In the example shown, the candidate pair agree on 'date of birth' (dob), 'last name' (sn) and 'first name' (fn), 'student status' (stud), 'sex', 'marital status' (marstat) and 'country of birth' (cob) but 'middle name' was missing (hence is not present on the x-axis) and there was disagreement on 'postcode' (pc) and 'industry code' (sic). Furthermore, the agreement on 'marital status' and 'country of birth' was on common responses and so did not add much weight to the match-score (final score). For 'residence type' (resi_type), both records had 'household'; although this is an agreement, a small negative weight was given as agreement on this value of this variable does not add to the likelihood that this pair of candidate records are duplicates. Overall, Splink waterfall charts demonstrate how the chosen candidate pair's score was derived, making Splink decisions easily traceable.

Figure 3.3-2
Waterfall chart showing the agreement weights awarded to a particular candidate pair of records



4. 2021 Census-to-census linkage as a case study

4.1 Variables

As the case study used census data, a wide variety of high-quality linkage variables were available. Those used for the linkage were ‘first name’, ‘middle name’, ‘last name’, ‘date of birth’, ‘sex’, ‘country of birth’, ‘residence type’ (communal establishment or household), ‘marital status’, ‘student status’, postcodes (‘home postcode’, ‘alternative postcode’, ‘one-year-ago home postcode’, ‘workplace postcode’) and ‘industry group’.

In addition, variables were derived for use in the blocking and case statements. For postcode variables, these included postcode area and postcode sector (e.g., for postcode PO15 5RR, area is PO and sector is PO15 5). For name variables, these included first and second parts of double-metaphone (Phillips, 2000), name initial, second character of the name and last character of the name.

4.2 Case statements

Separate case statements were written for ‘first name’, ‘middle name’, ‘last name’, ‘sex’, ‘date of birth’, ‘country of birth’, ‘residence type’, ‘marital status’, ‘student status’, postcodes and ‘industry group’. Trial and error was used to determine the most appropriate combinations and clauses. Splink visualisations were invaluable to facilitate this as we reviewed the scores obtained by pairs with different characteristics and errors i.e., ‘good’ matches, matches with errors and non-matches. A full list of the final case statements used in the case study is provided at (Xhaferaj, 2022).

For this particular project, separate case statements for ‘first name’, ‘middle name’ and ‘last name’ were used. However, in subsequent linkages of lower-quality data creating a single case statement combining all name variables was more effective. Users should experiment to find the best option for their dataset(s).

4.3 m-values: from the gold standard

The 23,434 confirmed duplicates in the gold standard were filtered using the case statements. m-values were calculated by dividing the number of records meeting each clause by the number of records with non-null values for that variable. The corresponding u-values were estimated using the ‘estimate_u_values’ Splink function.

4.4 Prior in the global model

The prior for the global model was estimated by dividing the proportion of duplicates in the gold standard by the total number of candidate pairs produced by a Cartesian join, $N(N-1)/2$, where N is the number of census records.

4.5 Blocking in the global model

The blocking strategy aimed to capture all of the true matches (duplicates) whilst minimising computational load. Two factors affect computational load: the total number of comparisons created by a blocking rule and the maximum number of comparisons in an individual block. For example, a rule using 'first name initial' and 'last name' might produce a feasible total number of comparisons. However, every pair with a given first name initial and last name combination would be placed in one block. If 10,000 people were called 'J Smith', this single block would contain 49,995,000 comparisons, which would cause a skew and make the computation inefficient. The Splink function 'get_total_comparisons' was used to check the total number of comparisons created by a rule. The Splink function 'column_combination_frequencies_chart' was used to check the size of individual blocks created by rules. Rules that created more than 10 million comparisons and/or at least one block with more than 100 people were re-written.

For most variables, if the value of the variable was missing on one or both of the records in a pair then they should not be selected when blocking using that variable. However, since it is meaningful to not have a middle name, the authors wrote a function to treat missing middle names as matches. This ensured persons with a missing middle name were not excluded from any blocks created using rules that included middle name.

Home postcodes are not necessarily expected to match when looking for duplicate census responses. It was therefore necessary to compare the different postcode variables to capture all potential duplicates. Twelve rules containing swapped postcode, e.g., 'home postcode' exactly matches 'alterative postcode', were included to account for this. Although it is intuitive to try and include the different combinations as OR statements e.g., 'home postcode' exactly matches 'alternative postcode' OR 'alternative postcode' exactly matches 'home postcode', we found that it was more computationally efficient to write separate blocking rules.

In total, 42 blocking rules were used, which together generated approximately 3 billion candidate pairs. The Splink global model took around 4 hours to run with this number of candidates. The global model blocking rules can be found at (Xhaferaj, 2022).

4.6 m-values: from Splink local models

Although the m-values created using the gold standard were used in the final global model, it was also important to demonstrate that similar values could be generated using the Splink local models. This functionality is necessary for future linkage projects where a gold standard is not available. Some experimentation was required to design suitable prior values and blocking for the local models so that reasonable m-values were calculated. Table 4.6-1 shows the prior values and blocking conditions used for each of the local models used in our final version.

Table 4.6-1
Description of the three local models used to generate m-values in Splink

Model	Prior	Blocking conditions
1	0.50	Exact match on: first name, middle name, last name, date of birth, sex
2	0.25	Exact match on: date of birth, sex, any postcode (home, alternative, one-year-ago, workplace)
3	0.50	Exact match on: first name, middle name, last name, any postcode (as in Model 2)

The m-values generated by each of the three local models can be found at (Xhaferaj, 2022). Using the same u-values as previously, the agreement weights were calculated. Splink's visualisations were used to support decisions regarding whether to use the harmonic mean m-values or the output from a single local model. For example, for marital status the m-values from Model 2 only were used, since Model 1, Model 3 and the harmonic mean gave a negative weight for agreement on 'marital status' of opposite-sex marriage. Although this is a common agreement status and therefore some down-weighting maybe expected, this amount of down-weighting seemed intuitively to be incorrect. Users should always sense-check the m values produced by the local models before using them in the global model.

A table showing the m-values and corresponding agreement weights for the gold standard method and the local models method for all of the case statements can be found at (Xhaferaj, 2022). The frequency distribution of match scores obtained using the Splink generated m-values was comparable to that obtained using the gold standard generated m-values.

5. Results

A visual assessment of the distribution of match scores produced by Splink demonstrated that it had successfully distinguished matches (i.e., duplicates) and non-matches. The distribution was bimodal with a large peak on the left, representing non-matches, and a much smaller peak on the right, representing matches. These peaks were clearly separable although, as is typical for probabilistic linkage, there was an area of overlap. This is illustrated in Figure 5-1: the large left-hand peak is clearly visible in the main graph whilst the small right-hand peak is visible in the inset. In Figure 5-2, the distribution of the scores for matches (duplicates) and non-matches is indicated. Matches are primarily in the right-hand peak, demonstrating that Splink has correctly identified the majority of true duplicates.

Figure 5-1
Distribution of match-scores generated by Splink using m-values generated from the gold standard dataset. The inset shows the same graph with a truncated y-axis

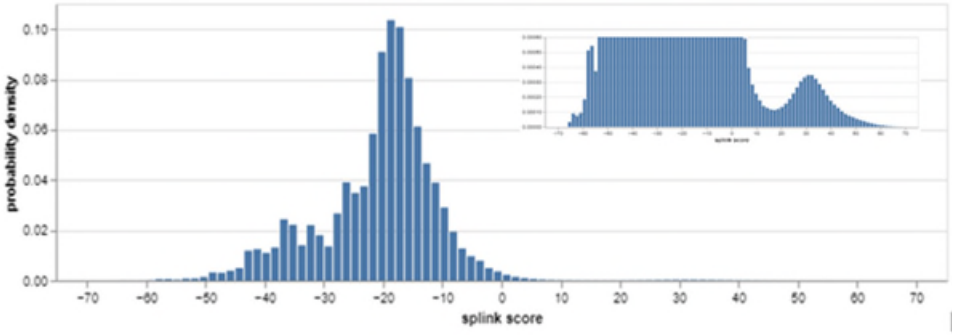
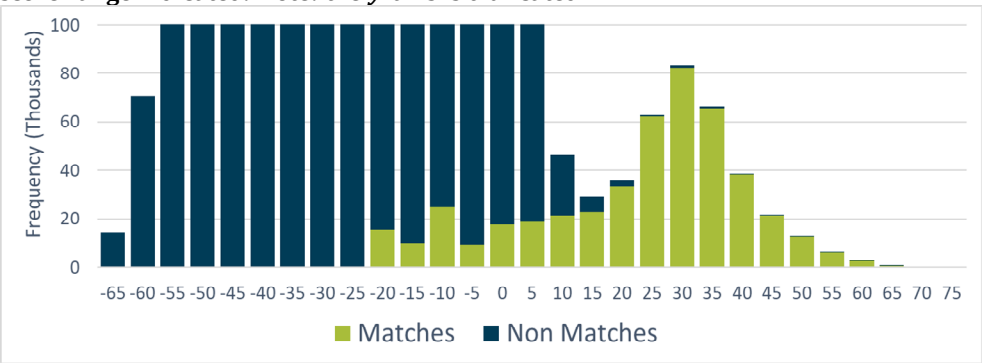


Figure 5-2
Distribution of match-scores generated by Splink with proportion of matches and non-matches for each score range indicated. Note: the y-axis is truncated



To quantify the ability of Splink to find duplicates in the entire Census (i.e., not just the gold standard), samples of 1,000 records and their candidate pair(s) were taken from 12 score ranges and clerically reviewed. The results demonstrate that, as expected, record pairs with a high match score were very likely to be duplicates and the probability of being a duplicate fell as the score decreased (Table 5-1). A slight exception was observed for scores (x) in the range $25 \leq x < 30$. No automatic linkage method is able to correctly assign all pairs; however, the small number of false positives occurring at the higher scores would likely be acceptable for most linkage projects.

Table 5-1
Percentage of sampled records confirmed as duplicates via clerical review

Match score (x)	Confirmed duplicates (%)	Match score (x)	Confirmed duplicates (%)
$40 \leq x$	99.45	$5 \leq x < 10$	14.90
$30 \leq x < 40$	98.80	$0 \leq x < 5$	3.47
$25 \leq x < 30$	98.90	$-5 \leq x < 0$	0.57
$20 \leq x < 25$	92.81	$-10 \leq x < -5$	0.44
$15 \leq x < 20$	77.46	$-15 \leq x < -10$	0.07
$10 \leq x < 15$	45.90	$x < -15$	0.00

6. Conclusions

The 2021 England and Wales Census was successfully linked to itself using Splink, enabling identification of duplicate census responses. The dataset contained approximately 58 million records. When joined using optimised blocking rules, approximately 3 billion candidate pairs were generated, which ran on the Splink global model in around 4 hours.

A gold standard dataset of duplicates and non-duplicates had already been created using the census data and thus could be used as training data for m-value calculation. However, the Splink local model implementation of the EM algorithm successfully estimated m-values that were comparable to those obtained from the gold standard.

The results of this case study support the assertion that Splink is computationally fast and methodologically accurate. The visualisations provided in Splink greatly facilitate fine-tuning of the algorithms, including writing blocking rules and case statements plus sense-checking parameters. In addition, users can view summaries of their results and use the waterfall graphs to provide algorithmic transparency at the individual record level.

The experience with Splink gained via this case study has shown that to work most effectively, Splink requires users who are knowledgeable about data linkage and understand how probabilistic linkage works. However, Splink is an excellent tool that guides the user though the entire process of probabilistic data linkage from blocking and setting parameters to outputting results. Splink has great potential for standardising data linkage across and between government departments and beyond. We recommend its use whenever probabilistic linkage is being contemplated.

References

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977), "Maximum Likelihood from Incomplete Data via the EM Algorithm", *Journal of the Royal Statistical Society*, 39(1), pp. 1-22.

Fellegi, I. P. and Sunter, A. B. (1969), "A Theory for Record Linkage", *Journal of the American Statistical Association*, 64(328), pp. 1183-1210.

Linacre, R. et al. (2022), *Splink.* [Online]
Available at: https://github.com/moj-analytical-services/splink, [Accessed 21 09 2022].

ONS (2022). *Coverage estimation for Census 2021 in England and Wales.* [Online]
Available at:
https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates/methodologies/coverageestimationforcensus2021inenglandandwales, [Accessed 11 11 2022].

Phillips, L. (2000), "The Double Metaphone Search Algorithm", *C/C++ Users Journal*, 18(6), pp. 38-43.

Shipsey, R. and White, Z. (2020), *Census to Census matching strategy 2021*, London: UK Statistics Authority.

Winkler, W. E. (1990), *String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage*, Washington DC: U.S. Bureau of the Census.

Xhaferaj, K. (2022), *ONS case-study of Splink use.* [Online]
Available at: https://github.com/Data-Linkage/Splink-census-linkage, [Accessed 02 11 2022].