Supervised network prediction for household statistics

Li-Chun Zhang[1,2]

[1] *University of Southampton, UK (L.Zhang@soton.ac.uk)*
[2] *Statistisk sentralbyrå, Norway*

Abstract

Producing census-like household statistics is possible based on relevant data originated from the administrative sources. However, address registration errors generally cause biased results. Under the assumption that households can be identified by survey at a sample of addresses, we develop a supervised network prediction approach to households at the out-of-sample addresses. An application to real data from the Norwegian Household Register is used to demonstrate the advantages of the proposed modelling approach over the existing practical alternatives.

Keywords: network modelling, community detection, graph representation learning, spectral embeddings, prediction accuracy

1 Introduction

Population censuses around the world are faced with challenges of increasing cost, declining response rates, and more frequent and rapid information needs. In contrast, some countries (especially in Scandinavia) have a long tradition of register-based census-like population statistics, which are not under the same pressures. There is thus naturally an ongoing shift towards greater reliance on administrative registers for producing census-like population statistics; see e.g. Bernardini et al. (2022) and Zhang (2022).

Countries such as Switzerland, Netherlands, Belgium and Slovenia have already adopted the register-based approach by 2010 (Skinner 2018). Various adjustments of available register data are considered by others. For example, Statistics Estonia (Tiit and Maasing, 2016) and Central Statistical Bureau of Latvia (CSBL, 2019) derive residency scores (between 0 and 1) for an Extended Population Register in order to produce head counts. Van der Heijden et al. (2022) use four registers to estimate the Māori population in New Zealand. Dunne and Zhang (2023) develop a system of population estimates compiled on administrative data only in Ireland.

Many other countries are engaged in their respective census transformation programmes, whereby administrative data are combined with coverage surveys to replace traditional censuses. For example, the Central Bureau of Statistics of Israel conducted a Register Survey for its last round of population census (Pfeffermann et al., 2019). Whereas the Italian National Institute of Statistics study audit sampling for quality evaluation of their register-based population statistics (Solari et al., 2023).

The Office for National Statistics in the UK is investigating a range of options, including register-based population counts (ONS, 2013 and 2017), as well as combining administrative registers and coverage surveys to replace population censuses (Law et al., 2023; ONS, 2023). See e.g. Brown and Murray-Close (2023) and Keller et al. (2018) for how the USA is researching incorporating administrative records into Decennial Census operations.

In this paper we consider a closely related problem that is less researched so far, which is the supervised learning approach to unbiased estimation of census-like household statistics based on administrative register data *and* a sample survey at selected addresses. In particular, we develop models that make use of household-related connections among the persons, which may be more efficient than models that do not directly use such connections.

Household statistics Rule-based approaches are used to derive households from relevant register data in several countries including all the Scandinavian countries. Auditing surveys for quality evaluation have been considered in the past; see e.g. Zhang (2011) and Axelson et al. (2021).

In this context household refers to one or several persons residing at the same address (or dwelling), which may be subject to additional conditions such as sharing housekeeping expenses or living arrangements. Mistaken *registered address* is therefore a main source of error of register-based households, which means the official, registered address of a person is not actually where the person resides, e.g. due to lack of updating or incorrect declaration to the registration office. Simply classifying all the persons with the same registered address as a household tend to result in too few households overall.

Table 1: Effect on household total given household registered at a resides at b

Other households registered	Other households registered	
	No b	Yes b
No a	Unaffected	Unaffected/Under-count
Yes a	Under-count	Under-count

This bias can be easily reasoned as follows. Suppose a household registered at address a actually resides at address b. Table 1 lists the effect on household total due to this registered address error, depending whether there are other households registered at addresses a or b.

- If there are no other households registered at these addresses, denoted by (No a, No b), the total number of households is unaffected.

- In case (No a, Yes b), the total is unaffected if there is only one household at b, since we still have two households altogether; whereas, if there are more than one household at b, then we have under-count of households.
- In case (Yes a, No b), the household total is under-counted.
- In case (Yes a, Yes b), there should be at least 3 households, so that we have under-count (by two households at these two addresses).

Thus, given all the registered addresses, one needs to determine whether the persons registered at the same address should nevertheless be classified into separate households so as to reduce the overall under-counting bias. Formally, this can be treated as a problem of *community detection among a network of connected persons*, where all the persons sharing the same registered address form a network and each community corresponds to a household.

A network classification approach has been developed from this perspective in Estonia (Visk et al., 2022). Roughly speaking, there are two major steps. First, a graph of person (as nodes) is constructed, where edges are introduced with associated weights that reflect the marginal probability of a pair of persons being in the same household. Second, community detection algorithms are implemented to delineate the households. Specifically, the Louvain method (Blondel et al., 2008) and Infomap (Rosvall et al., 2009) are used in a combined workflow. However, as we shall demonstrate later, these algorithms may lead to biased household statistics. Visk et al. (2022) noticed the issue as well and dealt with it practically by rule-based additional processing.

We shall adopt a different perspective, as classifying whether an apparent edge in the graph is *spurious*, which is the case if it connects two nodes that are not members of the same community. Taking advantage of a coverage survey, whereby the actual households are identified at a sample of addresses, we shall develop supervised learning to network models of the *target networks* (each corresponding to a community or household). Notice that any potential non-sampling errors in household identification are not considered in this paper, which is a practical issue for any sample survey or census. Notice also that any unsupervised node clustering method, such as the Louvain method or Infomap, is no longer effective from this perspective, since they cannot learn or improve from the observed target networks in the sample.

Related network methods Community detection in networks is of interest in many fields (e.g., Fortunato and Hric, 2016; Fortunato and Newman, 2022). The network connections may be induced by common location (e.g., address) or occupation (e.g., actors, researchers), communication, transport, social or economic interactions, biochemical reactions, and so on. Communities may exist among a network of connected nodes, if the nodes can be divided into non-overlapping groups which are more strongly connected (or similar) within each group than otherwise. Although the problem is vaguely defined in the unsupervised setting, since any partitions one makes are unvalidated, clarity of the target community exists for households provided they are identified by survey at sampled addresses.

Many community detection algorithms are based on optimisation according to a chosen modularity function, e.g. expressing how densely the connections are within the communities than otherwise. The Louvain method is based on this approach. Secondly, communities can also be created in terms of a dynamic process running on the graph, such as a random walk. Infomap is such a method. Thirdly, one can build a generative model of the networks, such as the stochastic block model (Holland et al., 1983), where the marginal probability that two nodes are connected depends solely on the communities they belong to. A better partition of nodes should yield a higher probability for the apparent networks, which can be used to find the best partition. However, the target communities remain unverified or latent.

Our proposal of supervised approach is more akin to graph representation learning (e.g. Hamilton, 2020). By representation learning of graphs one aims at ways to represent, or encode, the graph structure so that it can be exploited by machine learning models. Typically, a set of features, called *embeddings*, will be obtained from the given graph and possibly the values associated with the nodes or edges. One can derive the features for each node, each edge or the entire graph, using any appropriate representation learning methods.

For instance, node embeddings can be obtained by the so-called traditional methods, which do not require an explicit model of the target features, such as based on factorisation or random walks; see e.g. Khoshraftar and An (2022), Hamilton (2020). Whereas graph neural networks are increasingly being used for model-based node embeddings; see e.g. Zhou et al. (2020) for a review.

Figure 1 illustrates the well-known Zachary's karate club data (Zachary, 1977), where edges represent social relationships among the 34 members (i.e. nodes) and their subsequent split into two clubs are indicated by solid or circled nodes. As an example of node embedding, every node can be assigned one of four classes by modularity-based clustering (Brandes et al., 2008). Kipf and Welling (2017) show that this community structure can be closely captured by 2-vector node embeddings, using a 3-layer graph convolutional network, and these embeddings are comparable to those obtained by DeepWalk (Perozzi et al., 2014) which does not require explicit modelling.

Meanwhile, the x-coordinates of the nodes in Figure 1 are determined by an eigenvector related to the adjacency matrix of the graph. The details will be explained later as we develop our modelling approach. For the moment, simply notice that (a) this eigenvector amounts to a scalar embedding of each node, which is a reduction of the 34×34 adjacency matrix; (b) these scalar features can facilitate community detection, since the two groups of nodes can be perfectly classified by $\mathbb{I}(x_i < \eta)$, where $i = 1, ..., 34$, given any value $\eta \in (x_9, x_{10})$; (c) the value η can be estimated given a random sample of nodes with their associated x_i values and binary indicators of members split.

Our supervised network modelling approach will be similar to using x_i to predict the members split above, except that we shall operate at the network level rather than the node level. First, for each apparent network, we construct a graph using the relevant data, by introducing edges and associated weights indicating how likely a pair of nodes may belong to the same target network.

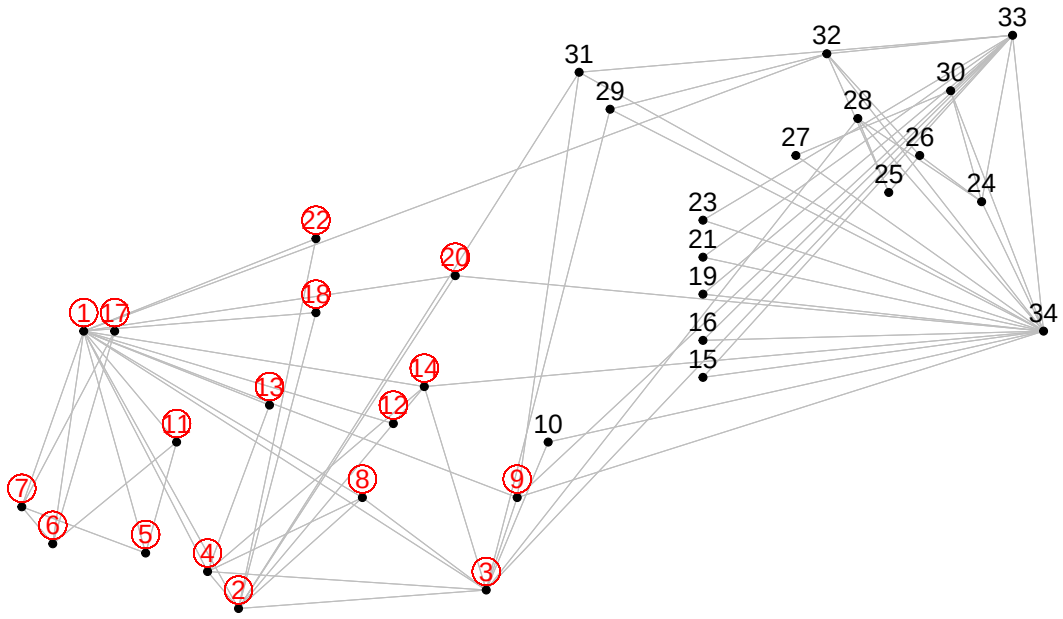


Figure 1: Members split (circled or solid) of Zachary's karate club

Next, we shall obtain a *graph-feature* vector for each apparent network which encode these network connections, say, x_a for apparent network a. Finally, we build predictive models for a given target variable observed in a sample of apparent networks, say, y_a given x_a and possibly other non-graph features.

In the context of households as target networks, an example of y_a may be the number of households at address a. Or y_a may be the number of edges between household members, which are a subset of all the edges in the apparent network since an apparent edge may be spurious if it connects persons in different households. Notice that in this paper we let each apparent network correspond to persons sharing the same registered address and each target network a household therein. This is the case in all the countries that produce register-based population *and* household statistics, since it is not possible to 'move' people to other addresses. However, the proposed network modelling approach is equally applicable, when an apparent network pertains to several households sharing several registered addresses, provided these are obtained by sample survey. For instance, suppose one person i_1 with registered address a and three persons (j_1, j_2, j_3) with address b, where (i_1, j_1) and (j_2, j_3) form two *de facto* households. Given the apparent network consisting of (i_1, j_1, j_2, j_3) and the two target networks, the network modelling approach is the same as will be described in this paper.

Rest of paper We develop the methods of supervised network prediction in Section 2, including the associated estimation of the out-of-sample prediction accuracy. Simulations will be used to illustrate the techniques involved and the efficacy of network models. In Section 3 we apply our approach to a large sample of households from the Norwegian Household Register, to investigate the likely gains over the existing practical alternatives, which are rule-based

classification, unsupervised community detection algorithms, and traditional predictive modelling without using the available graph structure in the data. Some final remarks and future topics will be given in Section 4.

2 Methods

The kind of graph data incorporating network connections that we will consider in this work can be formally described as follows.

Denote by $G = (U, A)$ an undirected simple graph with nodes U and edges A, where $(ji) \in A$ iff $(ij) \in A$ for $i \neq j \in U$, and $a_{ij} = 1$ if $(ij) \in A$ or $a_{ij} = 0$ otherwise. That is, the existence of edge (ij) from i to j implies that of (ji) in the opposite direction, and the adjacency matrix $[a_{ij}]$ is symmetric by definition. For undirected graphs it is thus convenient to read (ij) as the edge *between* the two nodes, rather than the directed edge from i to j *per se*.

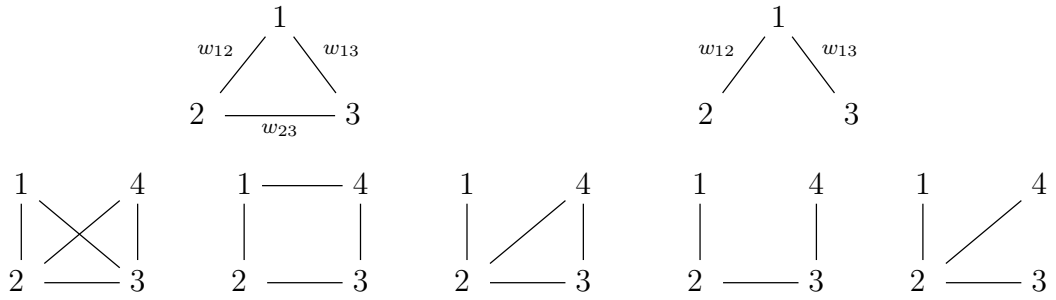


Figure 2: Top (left to right), triangle, 2-star; bottom (left to right), 2-triangle, 4-cycle, lollipop, 3-path, 3-star

We will refer to each component of G as an *apparent network* of connected nodes, denoted by $\kappa = 1, ..., N$. Figure 2 illustrates the smallest apparent networks with 3 or 4 nodes, which will be used for illustration in this section. However, suppose there may exist spurious edges in an apparent network, in the sense that $a_{ij} = 1$ may be the case even when i and j do not belong to the same *target network* of interest. Let c_i index the target network of node i, and let $\delta_{ij} = \mathbb{I}(c_i = c_j) = 1$ if $c_i = c_j$ for $i \neq j$ and $\delta_{ij} = 0$ otherwise. Each target network forms an own graph component according to

$$\Lambda = \{(ij) : \delta_{ij} = 1\} \quad \Leftrightarrow \quad c_U = \{c_i : i \in U\} \ .$$

One can either think of $G' = (U, \Lambda)$ as a graph associated with $G = (U, A)$, or one can think of $\mathbb{G} = (U, A \cup \Lambda)$ as a multigraph with two types of edges. Either way, the edges A are known, but the edges Λ are unknown initially.

Denote by $w_A = \{w_{ij} : (ij) \in A\}$ the weights assigned to the edges in the apparent networks, where $w_{ij} \equiv w_{ji}$ and $0 < w_{ij} \leq 1$. E.g., it may be possible to obtain w_{ij} as an estimate of the marginal probability $\Pr(\delta_{ij} = 1 \mid a_{ij} = 1)$, indicating how likely i, j belong to the same target network. Let the *weighted* adjacency matrix $[w_{ij}]$ have elements $w_{ij} \equiv 0$ if i, j are non-adjacent in G.

Example 1. *The top row of Figure 2 depicts two apparent networks, one triangle and one 2-star, with associated edge weights. Let us consider some relevant situations for household statistics.*

A triangle may represent three persons with the same registered address, where persons 1 and 2 are the parents of person 3. In this situation one would expect all the weights w_{ij} to be close or equal to 1, e.g. when these probabilities are estimated based on the last census.

A triangle may also represent three persons at a given address, where person 1 is the mother of person 3 whilst person 2 has no family relationship to them. The edges (12) and (23) are not spurious if e.g. persons 1 and 2 are unmarried partners, but they would be spurious if there is an address error and person 2 actually lives elsewhere, i.e. two households $\{1,3\},\{2\}$. In either case, one might expect the weights w_{12} and w_{23} to be lower than w_{13}.

In the case person 1 is the mother of person 3 and person 2 is an unrelated adult, one may choose not to introduce any edge (23) between the unrelated adult and child, such that the apparent network of these three persons is now a 2-star instead of triangle, because past experiences show that such three persons may form a household nearly always due to a partnership between the two adults.

In other words, how the edges are introduced in the apparent networks, i.e. the preparation of graph data, is a modelling choice from the beginning.

2.1 Community detection

As mentioned earlier, one may attempt to uncover the target networks using one of the many community detection methods; see e.g. Orman et al. (2012), Dao et al. (2020) for some comparisons. Denote the resulting target networks by $\hat{c}_U = \{\hat{c}_i : i \in U\}$, where $\hat{c}_j = \hat{c}_j$ iff nodes i, j belong to the same node partition resulting from the chosen algorithm.

In particular, the Louvain method aims to maximise

$$Q = \sum_{i,j \in U} \left(\frac{w_{ij}}{W} - \frac{1}{2} \frac{w_i}{W} \frac{w_j}{W} \right) \delta_{ij}$$

(Blondel et al., 2008), where $W = \sum_{i \in U} w_i$ and $w_i = \sum_{j \in U} w_{ij}$. Whereas Infomap aims to minimise the so-called map equation, which is equivalent to

$$L = \frac{W^-}{W} \log \frac{W^-}{W} - 2 \sum_c \frac{w_c^-}{W} \log \frac{w_c^-}{W} + \sum_c \frac{w_c^- + w_c}{W} \log \frac{w_c^- + w_c}{W}$$

(Rosvall et al., 2009), where $w_c = \sum_{i:c_i=c} w_i$, and $w_c^- = \sum_{i:c_i=c} \sum_{j:c_j \neq c} w_{ij}$ is the total weight of all the edges 'exiting' the cth cluster, such that $W^- = \sum_c w_c^-$ is the total weight of all the between-cluster edges.

However, as we shall demonstrate later on real data, the results of these methods are not satisfactory for household statistics. For a quick illustration, take the triangle in Figure 2: both these algorithms would always yield a single target network $\hat{c}_1 = \hat{c}_2 = \hat{c}_3$, regardless if the edge weights (w_{12}, w_{13}, w_{23}) are

$(1,1,1)$, $(0.75,1,0.01)$ or $(0.01,1,0.01)$. As we have discussed above, the last two sets of weights may correspond to some common situations where the three persons form two separate households due to address error.

2.2 Spectral embeddings

Since the mapping from (G, w_A) to $\hat{c}_U$ can be viewed as node embeddings, where $\hat{c}_i$ is a scalar feature of each node i, any community detection algorithm can also be viewed as a method of graph representation learning. But little is gained if $\hat{c}_U$ are largely equivalent to the apparent networks.

It is common to approach node embeddings through the adjacency matrix of the graph, such as the factorisation-based Laplacian eigenmaps technique of Belkin and Niyogi (2001). This is because the eigenvalues derived from the graph Laplacian matrix are closely related to the connectivity of the graph, and the set of eigenvalues (called spectrum) are invariant to how the nodes and the adjacency matrix are arranged, i.e., how the nodes happen to be named as 1, 2, 3, and so on; see e.g. Chung (1997).

Given the graph G of apparent networks with associated edge weights w_A, let M be the weighted graph Laplacian matrix, with elements

$$m_{ij} = \begin{cases} \sum_{l \in U} w_{il} & \text{if } i = j \\ -w_{ij} & \text{if } i \neq j \end{cases} .$$

Below we explain how to derive the embeddings of each apparent network κ directly from the corresponding sub-matrix of M. To avoid complicating the notation, we shall still use the notation M, instead of M_κ, and designate a pair of eigenvalue and eigenvector as λ and $z(\lambda)$, instead of λ_κ and $z_\kappa(\lambda_\kappa)$.

First, let M correspond to an apparent network. For a pair of eigenvalue λ and normalised eigenvector $z(\lambda)$ of M, we have

$$L(z; w) = \sum_{i<j:(ij)\in A} w_{ij}(z_i - z_j)^2 = z'Mz = \lambda z'z = \lambda$$

since $z'z = 1$ if z is normalised. Arranging the eigenvalues as $0 \leq \lambda_1 \leq \lambda_2 \leq \cdots$, which are all real and non-negative, we obtain the minimum $L(z; w) = 0$ only if the components of z are all equal to each other. It follows that

$$\lambda_1 = L(z(\lambda_1); w) = 0 \quad \text{and} \quad \lambda = L(z(\lambda); w) > 0$$

for any other $\lambda \neq \lambda_1$. Thus, if the probability of spurious edges in A is zero, then $c_i \equiv 1$ for each node i can simply be given as the eigenvector $z(\lambda_1)$ of M, due to the fact that the network is connected. Whereas, if $\Pr(\delta_{ij} = 0 \mid a_{ij} = 1) > 0$, i.e. the probability of spurious edges in A is non-zero, then one may consider the other eigenvectors of M, to be called the *spectral embeddings*, for which the corresponding eigenvalues are positive.

In particular, $z(\lambda_2)$ minimises $L(z; w)$ over the spectrum, where any two nodes i and j receive relatively more similar z-values the larger the edge weight

w_{ij} is between the two nodes. Such a vector encodes therefore both the varying strengths of connectivity as suggested by the edge weights and the structure of the apparent network in terms of its adjacency matrix.

Indeed, the x-coordinates in Figure 1 are simply given as $x_i = \text{rank}(z_i(\lambda_2))$, where $w_{ij} \equiv 1$. Using the rank instead of the value $z_i(\lambda_2)$ directly serves to space out the nodes evenly for a less cluttered view. Appendix A illustrates how the eigenvectors and eigenvalues vary intuitively with the edge weights for the two 3-node networks in Figure 2.

Next, given two apparent networks that are unconnected to each other, let M, λ, z be associated with the one of them and let M^*, λ^*, z^* be associated with the other. We have

$$\begin{pmatrix} M & 0 \\ 0 & M^* \end{pmatrix} \begin{pmatrix} z \\ 0 \end{pmatrix} = \lambda z \quad \text{and} \quad \begin{pmatrix} M & 0 \\ 0 & M^* \end{pmatrix} \begin{pmatrix} 0 \\ z^* \end{pmatrix} = \lambda^* z^* \ .$$

In this sense, the spectral embeddings of an apparent network are invariant of the other unconnected networks. Moreover, since this holds if $M^* = 0$, one can let the spectral embeddings to have a fixed dimension, by filling in 0's for networks with fewer nodes. For instance, let $(0.816, -0.408, -0.408, 0, ..., 0)$ be a q-vector for a triangle with 3 nodes, with $q - 3$ additional 0's given that the networks have at most q nodes in a given application.

2.3 Predictive modelling

For supervised learning, let $\mathbb{G}_s = (U_s, A_s \cup \Lambda_s)$ be a *labelled sample graph*, where U_s is a subset of nodes from U which correspond to the set of apparent networks according to A_s, as well as the set of target networks according to Λ_s, where $A_s \subset A$, $\Lambda_s \subset \Lambda$. That is, there does not exist any node outside U_s, which is connected to U_s via the edges in A or Λ. See Zhang (2021) for graph sampling methods which can arise in various situations. For this paper we shall simply assume a sample of addresses directly, which yields one apparent network at each address, given that our focus here is on the modelling approach.

For each apparent network κ, let y_κ be the outcome to interest (observed only in the sample), for which we shall create a feature vector x_κ, where some of the components of x_κ can be given by spectral embeddings as follows.

Denote by $z(\lambda_2), ..., z(\lambda_{n_\kappa})$ the eigenvectors of the weighted Laplacian matrix of this network, which correspond to the positive eigenvalues $\lambda_2 \leq \cdots \leq \lambda_{n_\kappa}$, where n_κ is the number of nodes in the network. One can choose any number of eigenvectors or eigenvalues to be included in x_κ. In particular, we would always include $z(\lambda_2)$ and λ_2, which tend to be the most useful. We apply also some additional transformation of the chosen eigenvectors.

- First, we normalise each eigenvector $z = (z_1, ..., z_{n_\kappa})'$ to be z^*, where

$$z_i^* = (z_i - \min_{1 \leq j \leq n_\kappa} z_j) / \sqrt{\textstyle\sum_{i=1}^{n_\kappa} (z_i - \min_{1 \leq j \leq n_\kappa} z_j)^2}$$

  for $i = 1, ..., n_\kappa$. The intuition is that the most useful features of z are the

distances between its components, as illustrated in Appedix A.

- Next, to remove the arbitrariness of node arrangement, i.e. which node happens to correspond to which row and column in the adjacency matrix, we rearrange $z^*(\lambda_2)$ in the increasing order, such that the first component is always 0, and rearrange $z^*(\lambda_j)$ accordingly, for each $j \neq 2$, such that the components of $z^*(\lambda_j)$ refer to the same nodes as the ordered $z^*(\lambda_2)$.
- Finally, given q as the maximum dimension of eigenvectors in an application, we complement $z^*(\lambda)$ with $q - n_\kappa$ zero's as the q-vector to be included in x_κ.

Of course, apart from these spectral embeddings, it is possible to include other features in x_κ, which are related to the nodes in κ. An example in the context of households may be the age of each person at a given address. We would arrange the persons in an age-vector in the same order as $z^*(\lambda_2)$.

2.4 An example

For a small example of supervised network modelling, let the graph contain the apparent networks in Figure 2.

Data We generate a sample consisting of $n = \sum_{t=1}^{7} n_t$ apparent networks, where n_t is the number of a given type of network, $t = 1, ..., 7$ in Figures 2. For any 3-node network, we generate (w_{12}, w_{13}, w_{23}) given the parameters $(\eta_{12}, \eta_{13}, \eta_{23})$, where

$$w_{ij} = 1 - \{\exp(u_{ij}) + 1\}^{-1} \quad \text{and} \quad u_{ij} \sim \text{Exponential}(\eta_{ij}) \ .$$

Similarly, for any 4-node network, we generate $(w_{12}, w_{13}, w_{14}, w_{23}, w_{24}, w_{34})$ given the corresponding parameters η_{ij}. Note that we would set $w_{ij} = 0$ for any edge that is absent in a given network, e.g. $w_{23} = 0$ for 2-star.

Given each apparent network, we simulate the target network edges $(ij) \in \Lambda$ independently conditional on $a_{ij} = 1$, where $\Pr(\delta_{ij} = 1 \mid a_{ij} = 1) = w_{ij}$. Different results can be generated. For instance, an apparent triangle can lead to a single target network (triangle or 2-star), or two target networks (i.e. a two-node clique and a separate node), or three separate single-node target networks.

The graph data are complete once we have generated the target edges for each apparent network $\kappa = 1, ..., n$.

Modelling For a given subgraph $\mathbb{G}_\kappa = (U_\kappa, A_\kappa \cup \Lambda_\kappa)$, with associated edge weights $w_{A_\kappa} = \{w_{ij} : (ij) \in A_\kappa\}$, we extract the minimum and the maximum positive eigenvalues λ_2 and λ_{n_κ}, the corresponding eigenvectors $z(\lambda_2)$ and $z(\lambda_{n_\kappa})$. The feature vector x_κ for each network has 10 $(= 4 + 4 + 2)$ components, using $z^*(\lambda_2)$ and $z^*(\lambda_{n_\kappa})$ as described previously. Let the predictive model target

$$r_\kappa = \sum_{i<j \in U_\kappa} \delta_{ij}$$

which is equivalent to the number of target network edges in Λ_κ.

Table 2: Empirical distribution of realised r_κ or modelled $\hat{r}_\kappa$

Setting		$r_\kappa = 0$	$r_\kappa = 1$	$r_\kappa = 2$	$r_\kappa = 3$	$r_\kappa = 4$	$r_\kappa = 5$
I	Realised	0.0502	0.1825	0.3170	0.3137	0.1170	0.0196
	Modelled	0.0465	0.1882	0.3210	0.3091	0.1155	0.0197
II	Realised	0.0219	0.1469	0.3018	0.3403	0.1585	0.0306
	Modelled	0.0198	0.1486	0.3092	0.3350	0.1566	0.0306
III	Realised	0.0198	0.1139	0.2175	0.3583	0.2231	0.0674
	Modelled	0.0170	0.1132	0.2267	0.3560	0.2205	0.0666
IV	Realised	0.0063	0.0610	0.2114	0.3900	0.2499	0.0815
	Modelled	0.0055	0.0574	0.2176	0.3907	0.2481	0.0808

Settings I to IV in Table 2 are created as we vary η_{ij} when simulating the edge weights, where the proportions of the different apparent networks may vary from about 5% to 30%. The dimension of η_{ij} is 3 or 6, respectively, for 3-node or 4-node networks, as given below (in 10^{-1}):

$$\text{I} : (5,5,5),(1,5,5,1,1,5); \quad \text{II} : (1,5,5),(1,5,5,1,1,1);$$
$$\text{III} : (1,5,5),(0.1,1,1,0.1,1,1); \quad \text{IV} : (1,1,1),(0.1,1,1,0.1,1,0.1).$$

Overall, the probabilities of larger r_κ increase gradually from setting I to IV, as can be seen from the empirical distribution of r_κ over all the networks, referred to as realised. As the total number of target network edges increases, the number of target networks would decrease as the apparent network becomes less likely to be fragmented according to Λ_κ.

Random forest models are fitted to $\{(r_\kappa, x_\kappa) : \kappa = 1, ..., n\}$ in all the settings. Denote by $\hat{r}_\kappa$ the predicted number of target network edges by the random forest model. The empirical distributions of $\hat{r}_\kappa$ are referred to as modelled in Table 2. The results are obtained with sample size $n = 3500$.

It can be seen that, across all the settings, spectral embeddings can yield essentially unbiased predictive model of r_κ, where the modelled proportions deviate from the realised ones only at the third digit level (i.e. 10^{-3}).

It may be emphasised that the models here use only features derived from the network connections, i.e. the edges and the associated weights, without any additional features at the node level or the network level. This shows that spectral embeddings can transform the network connections to useful features for supervised network prediction, just like we have seen earlier in the example of Zachary's karate club.

2.5 Prediction accuracy of random forest models

Zhang et al. (2025) develop unbiased mean squared error estimation of any predictor with respect to the sampling distribution, denoted by $s \sim f(s)$, where all the outcomes and features are treated as constants over repeated sampling. Such design-based predictive inference is valid whether or not the postulated or adopted model may be the true data model. Below we outline this approach

to prediction accuracy for binary outcomes when the model is random forest, where we let s be a simple random sample (SRS) of addresses as will be the case in our illustrative application later.

Denote by x_κ the feature vector specified for a random forest model of y_κ. Given any sample $s = \{1, ..., n\}$, let

$$\hat{p}_\kappa(s) = \hat{p}(x_\kappa, s) = \frac{1}{T}\sum_{t=1}^{T} \mu(x_\kappa, s_1^{(t)}) \tag{1}$$

be the random-forest probability of $y_\kappa = 1$ given x_κ, which is the average of the tree-probabilities in the forest, denoted by $\mu(x_\kappa, s_1^{(t)})$ for $t = 1, ..., T$, given T as the number of trees in the adopted model. The notation explicates $s_1^{(t)}$ as the t-th bootstrap sample (for the t-th tree-probability), which is selected from s by SRS with replacement. Let

$$s_2^{(t)} = s \setminus s_1^{(t)}$$

be the out-of-bag sample addresses which are not involved in $\mu(x_\kappa, s_1^{(t)})$, and let $n_2^{(t)}$ be the size of $s_2^{(t)}$. Notice that, as long as the sample s is selected without replacement, $s_2^{(t)}$ is SRS without replacement from $U \setminus s_1^{(t)}$, even though $s_1^{(t)}$ may contain duplications when it is selected from s with replacement.

Let the observed *training accuracy* of $\hat{p}_\kappa(s)$ be given as

$$\tau(s) = n^{-1}\sum_{\kappa \in s}\Big(y_\kappa \hat{p}_\kappa + (1 - y_\kappa)(1 - \hat{p}_\kappa)\Big) \tag{2}$$

and let its unobserved out-of-sample *prediction accuracy* be

$$\tau(U \setminus s) = (N - n)^{-1}\sum_{\kappa \in U \setminus s}\Big(y_\kappa \hat{p}_\kappa + (1 - y_\kappa)(1 - \hat{p}_\kappa)\Big)\ . \tag{3}$$

Let the observed *out-of-bag prediction error* of $\mu(x_\kappa, s_1^{(t)})$ over $s_2^{(t)}$ be

$$\tau_1(s_2^{(t)}) = \big(n_2^{(t)}\big)^{-1}\sum_{\kappa \in s_2^{(t)}}\Big(y_\kappa \mu(x_\kappa, s_1^{(t)}) + (1 - y_\kappa)\big(1 - \mu(x_\kappa, s_1^{(t)})\big)\Big)$$

As explained in Appendix B, an unbiased predictor of (3) is given as

$$\hat{\tau}(U \setminus s) = \frac{1}{T}\sum_{t=1}^{T} \tau_1(s_2^{(t)}) \tag{4}$$

i.e. we have $E\big(\hat{\tau}(U \setminus s)\big) = E\big(\tau(U \setminus s)\big)$ over repeated sampling $s \sim f(s)$.

3 An illustrative application

To apply supervised network prediction of household, we extracted a random sample of addresses and households from the Norwegian Household Register.

The *register households* are created by deterministic classification rules, which make use of the Population Register and other relevant administrative data such as family and registered partnerships. Yearly household statistics derived from this statistical register are available at `ssb.no`; see Zhang (2009, 2011) for the related approaches to quality evaluation.

For our illustrative application, we removed all the persons of age 0-16 who have at least one parent at the same address, because their household membership will be determined via their parents in practice. We treat all the remaining persons who share the same registered address as an apparent network, and the register households among these persons as the target networks.

Table 3: Summary of data at 367207 addresses

Address by no. persons					Address, ≥3 persons	
1	2	3	4	≥5	1 family	≥2 families
44.7%	41.6%	9.1%	3.5%	1.0%	29300	20631

	No. families at given address			
No. households	1	2	3	≥4
1, $y_\kappa = 0$	100%	33.6%	10.4%	4.6%
≥2, $y_\kappa = 1$	0%	66.4%	89.6%	95.4%

Table 3 provides an overview of the thus processed data at altogether 367207 addresses. The top half shows first the distribution of address by the number of persons at the address, i.e. 1, 2, 3, 4, and 5 or more (max 10 here), and then the number of families for all the addresses with 3 or more persons. The bottom half shows the prevalence of single or multiple households at a given address by the number of families at the address.

There is only one person at 44.7% of the addresses, each as a single-person household trivially. There are two persons at about 41.6% addresses, where the problem is whether a given pair of persons belong to the same household, which can be modelled without introducing a graph structure to the data. For the 49931 addresses with 3 or more persons, there is only one family at 29300 addresses where there is always only one household. For supervised network prediction of household, therefore, we shall focus on the addresses where there are 3 or more persons *and* two or more families.

At any such address κ, let $y_\kappa = 1$ if there are more than one household, and let $y_\kappa = 0$ if there is a single household at the address. The total of y_κ of 14957, and the proportion of $y_\kappa = 1$ among the 20631 addresses is

$$\theta = 0.725$$

which will be treated as the target parameter in our numerical reports later.

Given y_κ as the target outcome at address κ, relevant predictive features may be summary values at the address level, such as the number of persons or families at κ, or characteristics of the individuals, such as age, sex. Notice that although the number of families is clearly a useful feature for y_κ as can be

seen in Table 3, there are important differences between family and household, such as when an unmarried couple yielding two 1-person families but one 2-person household. Indeed, 1-person families counted for about 48% of all the families around the time of Census 2001 in Norway, whereas 1-person households counted for about 38% of all the households in the census, and register-based Family Statistics were discontinued shortly after Census 2001, because users found household statistics to be more relevant and useful.

In any case, utilising graph spectral embeddings of the apparent networks can potentially improve the predictive model, as described below.

3.1 Graph structure introduced

To obtain spectral embeddings as features at a given address, we need to define a valued graph to represent the apparent network at the address. We introduce the edges and their weights as follows.

First, let $(ij) \in A$ if i, j belong to the same family, and let $w_{ij} = 1$ simply. This is based on the 'experience' that family members at the same address almost never belong to separate households. As a result each family will form a clique (i.e. complete subgraph) within any apparent network.

Next, between any two cliques (of family), we introduce an edge between any two adults from different families. This reflects the 'experience' that different families may form a household largely due to non-marital adult relationships (such as partners), rather than relationships among the children in different families, or between an adult and a child in different families.

A note is needed regarding the terms 'child' and 'adult' here, which is not distinguished by an age threshold 18 *per se*. It is the practice of the Norwegian Population Register that a person is assigned on birth the same family ID as the mother. Most people will be assigned an own family ID later in life, typically when they move out of the childhood home to live on their own. In this context, anyone who is yet to obtain an own family ID for the first time is referred to as a child here (regardless age), and anyone else is an adult.

Note also that we have removed the children aged below 16 here, because these would have added little information for the target y_κ in this illustrative application, had one included them as nodes to the family-clique *and* assigned probability 1 to all additional edges. Since a clique is as strongly connected regardless the number of nodes in it, including children would cause additional variation in the dimension of the embedding vectors, without making the model more discriminative. But of course the choice could be different as the target variable changes, such as when y_κ is the number of single-parent households at a given address.

Now, for each edge (ij) between two adults from different families, we let the weights w_{ij} be the probability that the two adults form a household, where the probability is estimated from the addresses with only two persons from different families, using a logistic regression model with the following features of persons i and j:

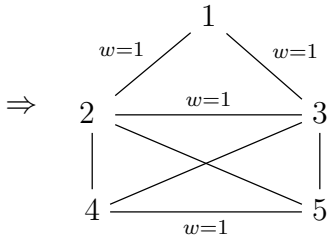
average age, absolute age difference, same sex (Yes, No), both male (Yes, No).

There are 44378 eligible pairs in this dataset for fitting the logistic model; the estimated probability has a minimum value 0.122, a maximum value 0.975, an average value 0.905 and a median value 0.952.

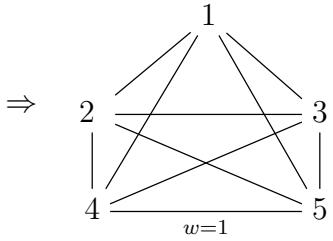
We notice that such a model of the *marginal* probabilities of pair-household is misspecified at addresses where there are more than two persons. However, since we only use these probabilities for spectral embeddings, what matters is how effective the resulting features are, but network prediction should not necessarily be biased because of it.

Table 4: Examples of apparent network, fictive Fam-ID and HH-ID

Node	Age	Sex	HH-ID	Fam-ID	Adult
1	16	1	156	545	No
2	53	1	156	545	Yes
3	55	0	156	545	Yes
4	79	0	166	384	Yes
5	78	1	166	384	Yes



Node	Age	Sex	HH-ID	Fam-ID	Adult
1	46	0	371	328	Yes
2	35	1	371	162	Yes
3	22	0	371	196	Yes
4	54	1	472	312	Yes
5	22	1	472	312	Yes



Example 2. *Table 4 illustrates the graph structure with two examples.*

In the first example, we have 5 persons at the given address from 2 families (as distinguished by the fictive Fam-ID). The first family consists of two adults and a child, the second family consists of a pair of adults. Each family forms a clique in the apparent network. In addition, edges $\{(24), (25), (34), (35)\}$ *are introduced, but not between the child 1 and the adult 4 (or 5) in a different family. The weights within each family is 1, indicated by '*$w = 1$*'; but less than 1 between adults from different families, omitted here to avoid cluttering the view.*

In the second example also with 5 persons at the given address, there are 4 families and no child at all. Consequently, the apparent network is a clique, where an edge exists between any pair of nodes. However, the edge weight is 1 only between adults 4 and 5 from the same family.

Notice that, had we included all the children aged below 16 as well, any clique of a family would have become larger where such children existed, but any clique of adults would have remained the same.

We have two households (as distinguished by the fictive HH-ID) at either the address. The households are quite typical in the first example: one of parents and a child, the other of an elderly couple. The households at the second address are less typical but easily imaginable. The HH-ID 472 may consist of an older adult and a younger relative, or a parent and an 'adult child' who moved back to the childhood home, and so on. A partner relationship may exist between the

adult 2 and one of the other two persons in HH-ID 371, but neither appears much more likely than the other.

The relative difficulty of household prediction in the second example can be seen in two respects. First, the whole graph forms a clique in the second example, in which case connectivity is intuitively stronger and more uniform through the apparent network (than otherwise). Second, the weight is 1 for $\{(12), (13), (23), (45)\}$ *or varies between 0.701 and 0.863 over* $\{(24), (25), (34), (35)\}$ *in the first example, whereas all the weights vary between 0.795 and 0.974 in the second example except* $w_{45} = 1$*, i.e. the between-family weights are more distinguished from the within-family weight (i.e. 1) in the first example.*

3.2 Methods applied

Supervised learning of any predictive model of y_κ will be based on SRS of the addresses, $s = \{1, ..., n\}$ from the population $U = \{1, ..., N\}$, where $n \leq N$ and $N = 20631$. Given the estimated probabilities $\hat{p}_\kappa$ of $\Pr(y_\kappa = 1)$ by a given model, the total of $y_\kappa = 1$ out of the sample will be estimated by summing $\hat{p}_\kappa$ over the out-of-sample addresses.

We compare a number of methods to understand the likely gains of the network prediction approach over the existing practical alternatives.

First, we will apply the Louvain method and Infomap as two examples of community detection algorithms. Since such unsupervised methods operate in the same way regardless the sample of y_κ-values, its expected accuracy (3) will be such that $E\big(\tau(s)\big) = E\big(\tau(U \setminus s)\big)$ with respect to SRS of s.

Second, the sample mean of y_κ is an unbiased estimator of $\theta = 0.725$. This corresponds to the *naïve* classifier $\hat{p}_\kappa \equiv 1$ practically with probability 1 given any moderate sample size n, which can be viewed as a simple family-based classification rule. With respect to SRS of s, its expected accuracy will be

$$E\big(\tau(s)\big) = E\big(\tau(U \setminus s)\big) = \theta \ .$$

Third, for supervised learning of predictive models, we consider three groups of features for y_κ:

- non-graph features: no. persons/families/families with children/adults;
- spectral embeddings of the weighted graph Laplacian submatrix: eigenvalue λ_2 and vector $z^*(\lambda_2)$, as described in Sections 2.2 and 2.3;
- a vector of ages (associated with the nodes), whose components are arranged in the same order as $z^*(\lambda_2)$ above.

To focus on the relative value of the graph features, we consider three models, all given as random forests, which use (i) only the 'non-graph' features, (ii) only the 'graph' spectral embeddings, and (iii) the 'full' set of features. Notice that, given q as the maximum dimension of $z^*(\lambda_2)$ to be allowed, the full model has $4 + 1 + q + q$ features. The prediction accuracy of any of these random forest models can be assessed as described in Section 2.5.

3.3 Results

First, applying the community detection algorithms to all the 20631 addresses, we obtain the proportion of $y_\kappa = 1$ to be 0.0059 by the Luovain method and 0.0004 by Infomap. Since the actual proportion is $\theta = 0.725$, classification by either of these algorithms is severely biased. This demonstrates clearly that such unsupervised methods are ineffective for the task of interest here.

Table 5: Training accuracy (2) of $\hat{p}_\kappa$ given $s = U$

Naïve	$\hat{p}_\kappa$	$1-\hat{p}_\kappa$	Non-graph	$\hat{p}_\kappa$	$1-\hat{p}_\kappa$
y_κ	0.725	0	y_κ	0.548	0.177
$1-y_\kappa$	0.275	0	$1-y_\kappa$	0.177	0.098
$\tau(s)$	0.725		$\tau(s)$	0.646	
Graph	$\hat{p}_\kappa$	$1-\hat{p}_\kappa$	Full	$\hat{p}_\kappa$	$1-\hat{p}_\kappa$
y_κ	0.594	0.130	y_κ	0.626	0.099
$1-y_\kappa$	0.127	0.148	$1-y_\kappa$	0.099	0.176
$\tau(s)$	0.743		$\tau(s)$	0.802	

Next, Table 5 gives the training accuracy (2) of the four predictive models, where the models are fitted to the all the addresses, i.e. the extreme case of $s = U$. The four decomposed terms of (2) are also given in Table 5, which are

$$\begin{pmatrix} \tau_{11}(s) & \tau_{10}(s) \\ \tau_{01}(s) & \tau_{00}(s) \end{pmatrix} = \begin{pmatrix} n^{-1}\sum_{\kappa\in s} y_\kappa \hat{p}_\kappa & n^{-1}\sum_{\kappa\in s} y_\kappa(1-\hat{p}_\kappa) \\ n^{-1}\sum_{\kappa\in s}(1-y_\kappa)\hat{p}_\kappa & n^{-1}\sum_{\kappa\in s}(1-y_\kappa)(1-\hat{p}_\kappa) \end{pmatrix}$$

where $\tau(s) = \tau_{11}(s) + \tau_{00}(s)$ and $1-\tau(s) = \tau_{10}(s) + \tau_{01}(s)$. The off-diagonal terms should be equal to each other in expectation, if the predictor $\hat{p}_\kappa$ is unbiased.

Clearly, although the training accuracy $\tau(s)$ of the naïve predictor $\hat{p}_\kappa \equiv 1$ is equal to θ, it is highly biased and the two off-diagonal terms 0 and 0.275 are totally imbalanced. The non-graph model is unbiased, but its training accuracy 0.646 is lower than the biased naïve predictor. The graph model is nearly unbiased, and its training accuracy 0.743 is an improvement over both the non-graph model and the naïve predictor. In particular, using only the network connections can lead to more accurate prediction than the model that does not use such connections. This demonstrates clearly the predictive power of graph data. Finally, the full model using all the features achieves a further improvement of the training accuracy (i.e. 0.802) in an unbiased manner.

Of course, in reality, one would prefer to obtain the model predictors based on a sample $s \subset U$, instead of a census if $s = U$. Table 6 shows the training accuracy of the full model at several reduced sample sizes n. The fitted full model is quite stable, where only the term $\tau_{00}(s)$ appears to decrease ever so slightly from 0.176 with $n = 20631$ to 0.166 with $n = 4133$.

Finally, Table 7 demonstrates the estimation of out-of-sample prediction accuracy (3) by (4). We generate 100 samples of addresses by SRS from U, where the sample size is fixed at $n = 4133$. Given each sample s, we obtain the predicted out-of-sample prediction accuracy by (4), and record the actual out-

Table 6: Full-model training accuracy (2) given sample size n

$n = 16488$	$\hat{p}_\kappa$	$1 - \hat{p}_\kappa$	$n = 12428$	$\hat{p}_\kappa$	$1 - \hat{p}_\kappa$
y_κ	0.624	0.101	y_κ	0.629	0.099
$1 - y_\kappa$	0.098	0.177	$1 - y_\kappa$	0.100	0.172
$\tau(s)$	0.801		$\tau(s)$	0.801	
$n = 8273$	$\hat{p}_\kappa$	$1 - \hat{p}_\kappa$	$n = 4133$	$\hat{p}_\kappa$	$1 - \hat{p}_\kappa$
y_κ	0.628	0.102	y_κ	0.628	0.103
$1 - y_\kappa$	0.102	0.168	$1 - y_\kappa$	0.103	0.166
$\tau(s)$	0.796		$\tau(s)$	0.794	

Table 7: Decomposed prediction of full-model prediction accuracy (3), based on 100 simple random samples each of 4133 addresses

Actual accuracy	$\hat{p}_\kappa$	$1 - \hat{p}_\kappa$	Predicted accuracy	$\hat{p}_\kappa$	$1 - \hat{p}_\kappa$
y_κ	0.6123	0.1126	y_κ	0.6117	0.1124
$1 - y_\kappa$	0.1087	0.1664	$1 - y_\kappa$	0.1096	0.1663

of-sample prediction accuracy by the random-forest predictor (1). We further decompose the predicted accuracy into the four terms $\hat{\tau}_{gh}(U \setminus s)$ for $g, h = 0, 1$, and decompose the actual accuracy into $\tau_{gh}(U \setminus s)$ for $g, h = 0, 1$. Table 7 gives the average of these decomposed terms over the 100 simulations. Clearly, unbiased estimation of prediction accuracy is achieved.

4 Final remarks

Producing census-like household statistics using administrative registers and a sample survey is a realistic option to be explored for census transformation in many countries around the world. We have developed network models that make use of the connections among individuals that naturally arise in this context. The associated method for unbiased estimation of the actual out-of-sample prediction accuracy is also developed.

In the presented analysis of data from the Norwegian Household Register, supervised network prediction is applied to 20631 addresses with three or more persons *and* two or more families. Due to random sampling, one can expect them to represent about 6% of the population, which is not a negligible part for official statistics. The proposed approach shows clear advantages compared to the existing practical alternatives, which include rule-based classification, unsupervised community detection algorithms, and predictive modelling without utilising the available graph structure in the data.

First, supervised network prediction is shown empirically to yield unbiased estimation, in contrast to rule-based methods which are always biased. Next, the Louvain method and Infomap for community detection cause a large bias; moreover, these algorithms cannot learn or improve from the observed target-network connections in sample surveys, which is a fundamental disadvantage

of such unsupervised methods. Finally, the proposed network models are more efficient than the models which do not directly use the information contained in the graph structure of the available data.

It should be noticed that, practically speaking, we do not exclude the possibility that rule-based household statistics may be considered 'fit-for-purpose' from a cost-benefit perspective in some countries despite the bias, and any more accurate supervised approach may be too costly for *statistical production* due to the need of sample survey. But even then, *quality evaluation* based on sample surveys may be necessary from time to time, for which it is always desirable to improve the efficiency of unbiased modelling approach.

As Ranalli (2025) shows in a recent review of Machine Learning methods for estimation in official statistics, the interest in advanced models has grown rapidly and the perception is no longer tenable that nothing beyond a linear regression model is truly necessary for official statistics. Although network models using graph features in the data may yet appear unfamiliar to many, it is intuitive that network connections may often contain useful information, so that there is no reason not to develop embedding methods to incorporate them in predictive modelling. Note that the spectral embeddings considered in this paper carry little extra computational cost; essentially, in addition to each non-graph feature vector, one only needs to perform a matrix eigendecomposition for the corresponding apparent network, whose dimension is quite small in the context of household networks.

A natural topic of future research is to enhance the features obtained by graph embeddings, which can incorporate additional connections among the individuals such as kinship derived from known parent-child data, as well as available indicators of the quality of registered address and individual-level features such as ethnocultural background. Better features should be able to improve the prediction accuracy in applications.

Another topic is to investigate the best possible classification methods, given the estimated household statistics and the learned network model, in order to produce statistical data at the household level, in addition to the sampled households. This may be of interest for various purposes such as economic or health analysis (with relevant study variables at the individual and household levels) or what-if analysis based on micro simulations.

A Examples of 3-node network embeddings

Table 8 illustrates how the eigenvectors and eigenvalues vary with the edge weights for the two 3-node networks in Figure 2.

The first case in Table 8 is a triangle with equal edge weights. Triangle is a complete subgraph, called a clique, where edge exists between any two nodes. When the edge weights are all equal, the non-zero eigenvalues of a clique are all the same, i.e. $\lambda_2 = \lambda_3$ here. Moreover, although which may be taken as $z(\lambda_2)$ or $z(\lambda_3)$ is indeterminant, the 'distance' between nodes 1 and 2 is equal to that between nodes 1 and 3 according to either the eigenvector.

Table 8: Examples of spectral embeddings for 3-node networks

Network, edge weight	Eigenvector : eigenvalue
Triangle, $w_{12} = w_{13} = w_{23} = 1$	$(0.816, -0.408, -0.408) : 3.000$ $(0.000, -0.707, 0.707) : 3.000$
2-star, $w_{12} = 1, w_{13} = 0.01$	$z(\lambda_2)' = (-0.405, -0.411, 0.816) : 0.150$ $z(\lambda_3)' = (0.709, -0.705, 0.004) : 2.005$
Triangle, $w_{12} = 1, w_{13} = w_{23} = 0.01$	$z(\lambda_2)' = (-0.408, -0.408, 0.816) : 0.030$ $z(\lambda_3)' = (0.707, -0.707, 0.000) : 2.010$
Triangle, $w_{12} = 1, w_{13} = w_{23} = 0.5$	$z(\lambda_2)' = (-0.408, -0.408, 0.816) : 1.500$ $z(\lambda_3)' = (0.707, -0.707, 0.000) : 2.500$
2-star, $w_{12} = w_{13} = 1$	$z(\lambda_2)' = (0.000, -0.707, 0.707) : 1.000$ $z(\lambda_3)' = (0.816, -0.408, -0.408) : 3.000$
Triangle, $w_{12} = 1, w_{13} = 0.99, w_{23} = 0.01$	$z(\lambda_2)' = (-0.004, -0.705, 0.709) : 1.015$ $z(\lambda_3)' = (0.816, -0.411, -0.405) : 2.985$
Triangle, $w_{12} = 1, w_{13} = 0.5, w_{23} = 0.01$	$z(\lambda_2)' = (-0.214, -0.576, 0.789) : 0.653$ $z(\lambda_3)' = (0.788, -0.579, -0.209) : 2.367$

All the three sets of edge weights in the second block of Table 8 suggest the possibility of two target networks, separating nodes $\{1, 2\}$ from 3. This is captured by the largely similar $z(\lambda_2)$ in all these cases, according to which nodes 1 and 2 are much closer to each other than to node 3. The other eigenvector $z(\lambda_3)$ is also largely similar, according to which node 1 is about as 'far' to 3 as node 2 to 3. The differences in the weights, i.e., $w_{13} = w_{23} = 0$ or close to 0 in the first two cases vs. $w_{13} = w_{23} = 0.5$ in the third case, are reflected in the eigenvalues (λ_2, λ_3), which are about $(0, 2)$ in the first two cases vs. $(1.5, 2.5)$ in the last case, i.e. eigenvalues are useful in addition to the eigenvectors.

The edge weights of the two networks in the next block of Table 8 suggest intuitively similar likelihoods of possible target networks, which is well captured by the closeness of their respective eigenvectors and eigenvalues.

The edge weights in the last case in Table 8 suggest that nodes 1 and 2 are closest to each other, regardless whether node 3 may be separated from them, which is captured by the eigenvector $z(\lambda_2)$, where $z_1(\lambda_2)$ and $z_2(\lambda_2)$ are closer to each other than to $z_3(\lambda_2)$, and $z_1(\lambda_2)$ is closer to $z_3(\lambda_2)$ than $z_2(\lambda_2)$ to $z_3(\lambda_2)$.

B Unbiased prediction of $\tau(U \setminus s)$

Now that $s_2^{(t)}$ is SRS without replacement from $U \setminus s_1^{(t)}$, $\tau_1(s_2^{(t)})$ is an unbiased predictor of the prediction error of $\mu(x_\kappa, s_1^{(t)})$ conditional on $s_1^{(t)}$, denoted by

$$\tau_1(U \setminus s) = (N-n)^{-1} \sum_{\kappa \in U \setminus s} \Big(y_\kappa \mu(x_\kappa, s_1^{(t)}) + (1 - y_\kappa)\big(1 - \mu(x_\kappa, s_1^{(t)})\big) \Big)$$

i.e.

$$E_s\Big(\tau_1(s_2^{(t)}) \mid s_1^{(t)}\Big) = E_s\Big(\tau_1(U \setminus s) \mid s_1^{(t)}\Big)$$

where the expectation is with respect to $f(s \mid s_1^{(t)})$, given

$$f(s)f(s_1^{(t)} \mid s) = f(s_1^{(t)})f(s \mid s_1^{(t)})$$

(Zhang et al., 2025). Apply the same argument to $t = 1, ..., T$, we obtain

$$\begin{aligned} E_s\Big(\frac{1}{T}\sum_{t=1}^{T}\tau_1(s_2^{(t)}) \mid s_1^{(1)}, ..., s_1^{(T)}\Big) &= E_s\Big(\frac{1}{T}\sum_{t=1}^{T}\tau_1(U \setminus s) \mid s_1^{(1)}, ..., s_1^{(T)}\Big) \\ &= E_s\Big(\tau(U \setminus s) \mid s_1^{(1)}, ..., s_1^{(T)}\Big) \end{aligned}$$

where the last equality follows from the definitions (1) and (3).

Data availability
The code can be found on the GitHub page at `https://github.com/statistikkon/supervised-network-prediction`

References

[1] Axelson, M., Holmberg, A., Jansson, I. and Westling, S. (2021). A Register-Based Census: The Swedish Experience. Chapter 8, pp. 181-204. In *Administrative Records for Survey Methodology*, eds. A. Y. Chun, M. Larsen, G. Durrant and J.P. Reiter. John Wiley & Sons, Inc.

[2] Belkin, M. and Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, 14:586-691.

[3] Bernardini, A., Brown, J., Chipperfield, J., Bycroft, C., Chieppa, A., Cibell, N., Dunnet, G., Hawkes, M.F., Hleihel, A., Law, E.C., Ward, D., and Zhang, L.C. 2022. "Evolution of the Person Census and the Estimation of Population Counts in New Zealand, United Kingdom, Italy and Israel. "*Statistical Journal of the IAOS*, 38(4):1221-1237. DOI: 10.3233/SJI-220018

[4] Blondel, V.D., Jean-Loup, G., Renaud, L. and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment.* `doi:10.1088/1742-5468/2008/10/P10008`

[5] Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, M., Nikoloski, Z. and Wagner, D. (2008). On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20(2):172-188.

[6] Brown, D. and Murray-Close, M. (2023). *Producing U.S. Population Statistics Using Multiple Administrative Sources*. Working Papers 23-58, Cen-

ter for Economic Studies, U.S. Census Bureau. https://www.census.gov/library/working-papers/2023/adrm/CES-WP-23-58.html

[7] Chung, F.R.K. (1997). *Spectral Graph Theory.* Providence, RI: American Mathematical Society.

[8] CSBL (2019). *Method Used to Produce Population Statistics*. Central Statistical Bureau of Latvia. https://www.csb.gov.lv/sites/default/files/data/15_04_2019_Iedz_Metodologija_ENG.pdf

[9] Dao, V-L, Bothorel, C. and Lenca, P. (2020). Community structure: A comparative evaluation of community detection methods. *Network Science*, 2020, 8:1-41. doi:10.1017/nws.2019.59

[10] Dunne J. and Zhang L.-C. (2023). A System of Population Estimates Compiled from Administrative Data Only (with Discussions). *Journal of the Royal Statistical Society, Series A*, 187: 3-38.

[11] Fellegi, I.P. and Sunter, A.B. (1969). A theory for record linkage. *Journal of the American Statistical Association*, 64:1183-1210.

[12] Fortunato, S. and Newman, M.E.J. (2022). 20 years of network community detection. *Nature Physics*, 18:848-850.

[13] Fortunato, S. and Hric, D. (2016). Community detection in networks: A user guide. *Physics Reports*, 659:1-44.

[14] Hamilton, W.L. (2020). Graph Representation Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Vol. 14, No. 3 , pp. 1-159. Morgan and Claypool.

[15] Holland, P.W., Laskey, K.B. and Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social Networks*, 5:109-137.

[16] Keller, A., Mule, V. T., Morris, D. S., and Konicki, S. (2018). A Distance Metric for Modeling the Quality of Administrative Records for Use in the 2020 U.S. Census. *Journal of Official Statistics*, 34:599-624.

[17] Khoshraftar, S. and An, A. (2022). A Survey on graph representation learning methods. https://doi.org/10.48550/arXiv.2204.01855

[18] Kipf, T. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. https://doi.org/10.48550/arXiv.1609.02907

[19] Law, A., Large, A., Hammond, C. and Linton, M. (2023). *Population stock estimates using linked administrative data and a coverage survey – a case study for 2021 and future directions.* ons.gov.uk

[20] Lee, D., Zhang, L.-C. and Kim, J.K. (2022). Maximum entropy classification for record linkage. *Survey Methodology*, 48:1-23.

[21] Office for National Statistics (2023). *Dynamic population model, improvements to data sources and methodology for local authorities, England and Wales: 2021 to 2022*. ons.gov.uk

[22] Office of National Statistics (2017). *Research Outputs: Coverage-adjusted administrative data population estimates for England and Wales, 2011*. ons.gov.uk

[23] Office of National Statistics. 2013. *Beyond 2011: Producing Population estimates Using Administrative Data: In Theory*. ons.gov.uk

[24] Orman, G., Labatut, V. and Cherifi, H. (2012). Comparative evaluation of community detection algorithms: a topological approach. *Journal of statistical mechanics: Theory and experiment*, P08001. doi:10.1088/1742-5468/2012/08/P08001

[25] Perozzi, B., Al-Rfou, R. and Skiena, S. (2014). DeepWalk: Online learning of social representations. https://doi.org/10.48550/arXiv.1403.6652

[26] Pfeffermann, D., Ben-Hur, D. and Blum, O. (2019). Planning the next census for Israel. Statistics in Transition, 20:7-19.

[27] Ranalli, M.G. (2025). Machine learning methods for estimation in Official Statistics. *Journal of Official Statistics*, 41:912-920.

[28] Rosvall, M., Axelsson, D. and Bergstrom, C.T. (2009). The map equation. https://doi.org/10.48550/arXiv.0906.1405

[29] Skinner, C.J. (2018). Issues and Challenges in Census Taking. *Annual Review of Statistics and Its Application* 5(1): 49-63. DOI: https://doi.org/10.1146/annurev-statistics-041715-033713

[30] Solari F., Bernardini A. and Cibella N. (2023). Statistical framework for fully register based population counts. *METRON*, 81:109-129. https://doi.org/10.1007/s40300

[31] Tiit, E.-M. and Maasing, E. (2016). Residency index and its applications in censuses and population statistics. Eesti statistika kvartalikri. (Quarterly Bulletin of Statistics Estonia). 3/16:41-60. http://www.stat.ee/publication-2016_quarterly-bulletin-of-statistics-estonia-3-16

[32] Van der Heijden, P.G.M., Cruyff M., Smith P. A., Bycroft C., Graham P., Matheson-Dunning, N. (2022). Multiple system estimation using covariates having missing values and measurement error: estimating the size of the Māori population in New Zealand. *Journal of the Royal Statistical Association, Series A*, 185:156-177.

[33] Visk, H., Levenko, V., Lehto, K., Maasing, E. and Tiit, E.-M. (2022). Households and dwellings for register-based census: a graph-based approach. *Baltic-Nordic-Ukrainian Network on Survey Statistics Workshop on Survey Statistics, 2022, Tartu.*

[34] Zachary, W. W. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452-473.

[35] Zhang, L.C. (2022). "Complementarities of Survey and Population Registers". *Statistics Reference Online*, Wiley. https://onlinelibrary.wiley.com/doi/10.1002/9781118445112.stat08352

[36] Zhang, L.-C. (2021). *Graph Sampling*. CRC Press.

[37] Zhang, L.-C. (2011). A unit-error theory for register-based household statistics. *Journal of Official Statistics*, 27:415-432.

[38] Zhang, L.-C., Sanguiao-Sande, L. and Lee, D. (2025). Design-based predictive inference. *Journal of Official Statistics*, 41:404-432.

[39] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C. and Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, 1:57-81.