Projecte de Neo4j: Padrons

Introducció: Padrons

Un grup d'investigadores en demografia històrica i reconeixement de documents vol estudiar l'evolució socio-econòmica d'una població a partir de la informació dels padrons de poblacions. Els padrons són els llistats d'habitants que elabora un municipi on figura la informació la seva informació com noms, cognoms, edat i altres dades personals. En l'actualitat aquesta informació es manté actualitzada constantment gràcies als sistemes informàtics però antigament es recopilava manualment cada 3-5 anys aproximadament. Aquesta informació es registrava en llibres com el que es pot veure a la Figura 1.

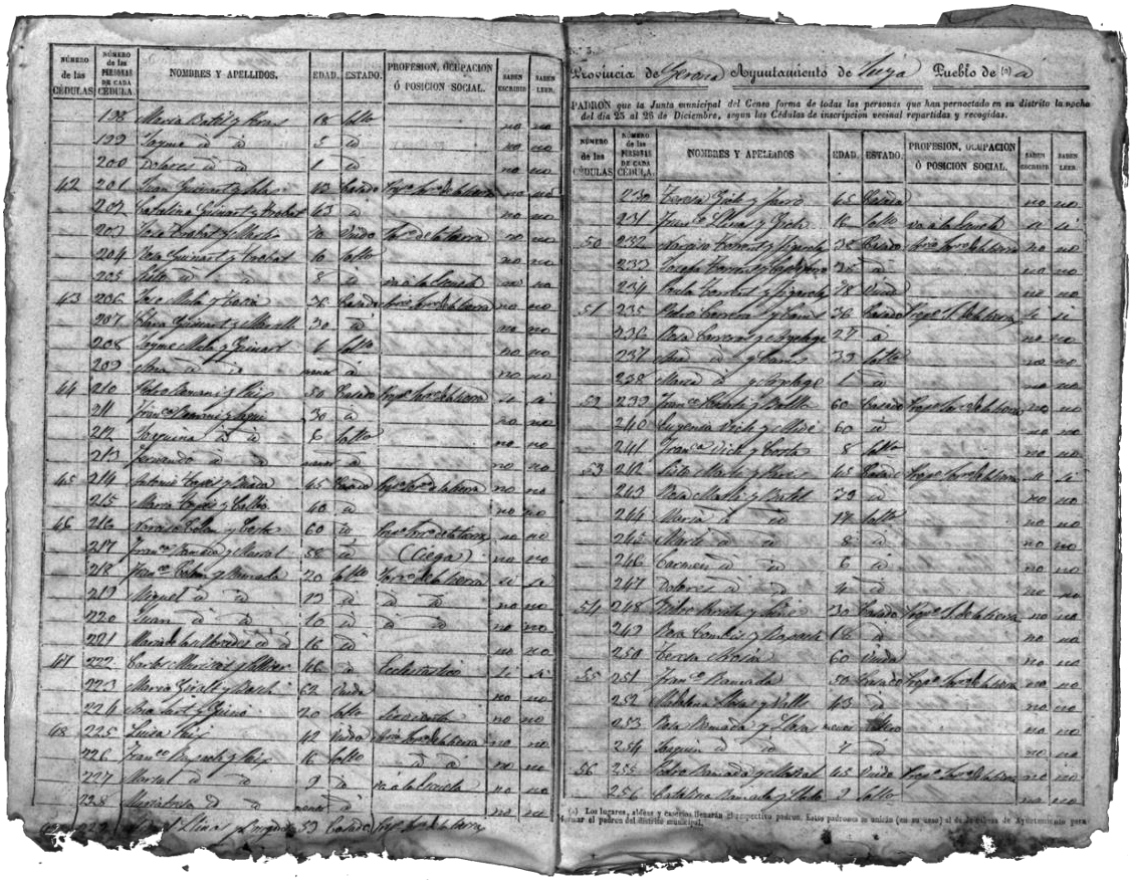

Figura 1. Padró del Cens de població amb les persones que pernoctaren al municipi de Julià el 25 de desembre de 1860.

La informació d'aquests documents, un cop digitalitzats, es processen amb tècniques de visió per computador i s'organitzen en una base de dades relacional. Per construir la base de dades, s'ha extret, de cada padró, la informació de cada habitatge, les persones que hi viuen i la relació de parentesc que hi ha entre ells. De cada habitatge, guarden la adreça completa (carrer, numero, pis), codi postal, barri i població. De cada persona que viu a l'habitatge, es guarda quina relació de parentesc el vincula amb el/la cap de família. A més, necessiten saber la ocupació (si treballa), una estimació dels ingressos (bruts) anuals de cadascun d'ells i l'estat civil. A més, com que els enregistraments poblacionals es repeteixen cada 3-5 anys la informació que es recopila per cada habitatge es va repetint per aquelles famílies que viuen en els mateixos habitatges.

Interessa identificar la informació dels individus al llarg del temps. Aquesta identificació es fa essencialment a partir de les dades personals (nom i cognoms) però en ocasions no és suficient. En aquests casos s'utilitza altres dades, com la data de naixement (inferida a partir de l'edat i l'any de padró), l'ofici o les relacions familiars. Tota aquesta informació s'ha organitzat segons el disseny Entitat-Relació de la Figura 2.

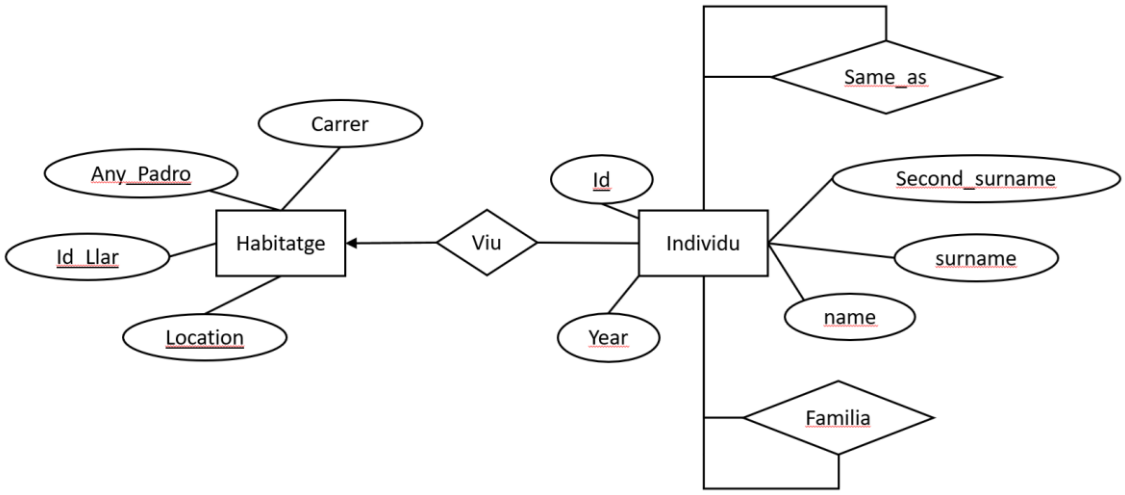

Figura 2. Disseny Entitat-Relació

Aquesta representació permet fer cerques i mostrar-les via una interfície web (http://dag.cvc.uab.es/xarxes/), però no permet analitzar fàcilment les relacions entre els individus al llarg del temps en els diferents habitatges en el que han viscut. Per això es vol representar la informació mitjançant una base de dades no relacional basada en grafs.

Projecte

En aquest projecte es treballarà, a partir d'un subconjunt de dades de padrons, la càrrega de dades i consultes en una base de dades de grafs (Neo4J) i es practicarà l'ús d'algorismes d'analítica de grafs.

Material

Disposeu del següent material per a realitzar el projecte:

- Fitxers de dades (CSVs).
- Script per visualitzar-les.

Treball previ

Abans d'iniciar el projecte caldrà llegir i visualitzar tota la documentació del projecte. A més cada grup haurà de crear un repositori privat de codi a **github** i donar accés al professorat que tutoritza el treball: alicia.fornes@uab.cat

El repositori s'anomenarà de la següent manera: **Neo4j_<NIU>** on NIU serà el NIU d'un dels integrants de l'equip de treball.

Projecte - Part 1.

Suposem que representem la base de dades no relacional basada en grafs de la Figura 3, que conté nodes de tipus *habitatges* i *individus*, i tres tipus de relacions:

- *viu*: representen el lloc on viu cada individu.
- *família:* relacions de parentesc entre individus que conviuen al mateix habitatge.
- *same_as*: els nodes que representen el mateix individu al llarg del temps.

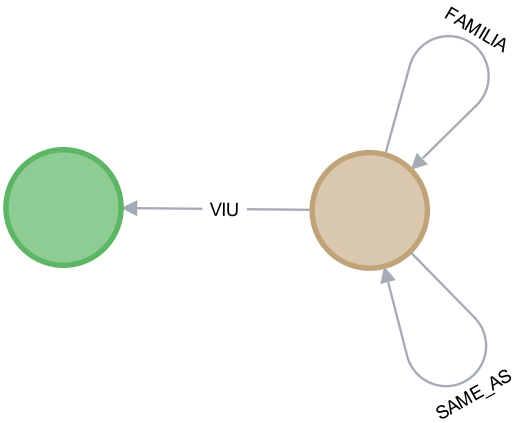

Figura 3. Esquema del graf de padrons

Exercici 1. Importeu les dades en la BD de Neo4j del projecte.

Genera un script en cypher que carregui totes les dades, generi tots els nodes, relacions i afegeixi les característiques allà on toqui. Consideracions:

- Feu servir *constrains* i *indexos* quan sigui necessari.
- Assegureu-vos que en executar el script dues vegades no es dupliquin les dades (https://neo4j.com/docs/cypher-manual/current/constraints/).
- No carregueu files *null* del fitxer CSV (*Id* de municipi, llar o individu = *null*).
- Feu les conversions de tipus que siguin necessàries.

Exercici 2. Resoleu les següents consultes Cypher:

a) Per a cada padró (any) de Sant Feliu de Llobregat (SFLL), retorna l'any de padró, el número d'habitants, i la llista de cognoms. Elimina duplicats i "nan".

b) Retorna totes les aparicions de "miguel estape bofill". Fes servir la relació SAME_AS per poder retornar totes les instancies, independentment de si hi ha variacions lèxiques (ex. diferents formes d'escriure el seu nom/cognoms). Mostra la informació en forma de taula: el nom, la llista de cognoms i la llista de segon cognom (elimina duplicats).

c) Mostra els fills o filles (només) de "benito julivert". Mostra la informació en forma de taula: el nom, cognom1, cognom2, i tipus de relació. Ordena els resultats alfabèticament per nom.

d) Mostreu les famílies de Castellví de Rosanes amb més de 3 fills. Mostreu el nom i cognoms del cap de família i el nombre de fills. Ordeneu-les pel nombre de fills fins a un límit de 20, de més a menys.

e) Per cada padró/any de Sant Feliu de Llobregat, mostra el carrer amb menys habitants i el nombre d'habitants en aquell carrer. Fes servir la funció *min()* i CALL per obtenir el nombre mínim d'habitants. Ordena els resultats per any de forma ascendent.

NOTA. Per a cada consulta incloeu el codi i el resultat obtingut.

Exercici 3. Analítica de Grafs:

Analitzeu les dades del graf per entendre millor l'estructura de les dades.

a) Estudi de les components connexes (cc) i de l'estructura de les components en funció de la seva mida. Feu servir el mode stream. Un cop calculades les components connexes (nodes *individu*, *habitatge* i relació *VIU*), feu **dues** consultes per explorar les dades. Per exemple (podeu fer-ne d'altres):

- Mostra, en forma de taula, les 10 components connexes més grans (ids i mida).
- Per cada municipi i any el nombre de parelles del tipus: (Individu)—(Habitatge).
- Quantes components connexes no estan connectades a cap node de tipus 'Habitatge', és a dir, els individus sense casa.

b) Semblança entre els nodes. Ens interessa saber quins nodes són semblants com a pas previ a identificar els individus que són el mateix (i unirem amb una aresta de tipus SAME_AS). Abans de fer aquest anàlisi:

- Determineu els habitatges que són els mateixos al llarg dels anys. Afegiu una aresta amb nom "MATEIX_HAB" entre aquests habitatges. Per evitar arestes duplicades feu que la aresta apunti al habitatge amb any de padró més petit.
- Creeu un graf en memòria que inclogui els nodes Individu i Habitatge i les relacions VIU, FAMILIA, MATEIX_HAB que acabeu de crear.
- Calculeu la similaritat entre els nodes del graf que acabeu de crear, escriviu el resultat de nou a la base de dades i interpreteu els resultats obtinguts.

NOTA. Acompanyeu cada consulta d'una explicació dels resultats.

Projecte - Part 2 (optatiu)

Exercici 4. Comparativa sobre diferents esquemes de base de dades:

Dissenyeu un altre esquema de la base de dades no relacional basada en grafs. En concret:

a) Dibuixeu l'esquema proposat. En concret, ha de tenir nodes de mínim 3 tipus (és a dir, no només *labels* de tipus *habitatges* i *individus)*.

b) Importeu les dades.

c) Feu 2 consultes Cypher (no cal que siguin consultes fetes a l'exercici 2) emprant el nou esquema proposat. Escriviu el codi i el resultat obtingut.

d) Feu aquestes mateixes 2 consultes, però ara segons l'esquema de la Figura 3.

e) Raoneu els avantatges i inconvenients de l'esquema proposat en comparació al model proposat a la Figura 3.

Presentació del Projecte

La presentació del projecte serà **obligatòria** per als grups que facin la part 2 del projecte. A la presentació es farà l'exposició de l'exercici 4 exclusivament. Per a la presentació oral es pot fer servir *powerpoint* o similar. La durada de l'exposició serà d'uns 10min. Tots els membres de l'equip han de ser presents i participar a l'exposició oral.

Material a lliurar

Haureu de lliurar un informe on s'expliqui raonadament la resolució de cada exercici. Aquest informe haurà de ser auto contingut i contenir una secció: treball en equip, on s'expliqui la distribució de tasques entre els components de l'equip i qui s'ha responsabilitzat de cadascuna. Cada membre s'haurà de responsabilitzar d'almenys d'una tasca. La distribució de tasques haurà de ser consistent amb els commits al repositori del projecte.

A l'informe s'haurà d'indicar l'adreça del repositori que haurà de contenir tot el material generat per l'informe. Això inclou tant els scripts en Cypher per importar les dades com els scripts per resoldre els altres exercicis.

Puntuació

La puntuació del projecte serà el següent:

Exercici	Puntuació
Exercici 1	2
Exercici 2	2
Exercici 3	3
Exercici 4	3

Factors multiplicatius de la nota.

Hi haurà dos factors multiplicatius que s'aplicaran globalment a la nota final del projecte. Els factors multiplicatius són:

- Treball en equip i ús del repositori de codi. Aquest factor s'aplicarà per avaluar el treball en equip i es podrà aplicar factors diferents als membres de l'equip. Normalment s'aplicarà un factor de 1 si el grup ha funcionat normalment. En cas que hagi evidències que algun membre de l'equip no faci la seva part se li podrà aplicar fins a un factor de 0 a nivell individual.
- Qualitat de l'informe i de la presentació del projecte. Aquest factor s'aplicarà per avaluar aspectes globals de la presentació de l'informe i també de l'exposició oral. Una presentació molt deficient de la feina feta podrà ser motiu suficient per suspendre el projecte.