

# PROJECTE DE NEO4J: PADRONS

---

Enginyeria de Dades

Base de Dades no Relacionals

Curs 2021/2022

2n semestre

Sergio Trigueros – NIU: 1605983

Daniel Pauli – NIU: 1568073

Abril Piñol Chacon – NIU: 1604159

Xavier Querol Bassols – NIU: 1526451

## Adreça del repositori

[https://github.com/danielpaulisch/projecte\\_neo4j\\_grup\\_82\\_3](https://github.com/danielpaulisch/projecte_neo4j_grup_82_3)

### Exercici 1

Per importar les dades en la BD de Neo4j del nostre projecte hem fet servir les consultes que adjuntem a continuació. Per començar creem dos restriccions, una que evita crear dos nodes del tipus Persona amb el mateix ID i una que evita que puguem crear dos nodes Habitatge amb el mateix valor en la propietat *Unique*, que és un identificador. Aquestes restriccions són necessàries si executem l'script més d'una vegada ja que d'aquesta manera no acabarem tenint tots aquests nodes duplicats.

Quan ja tenim les restriccions definides comencem a llegir les dades dels fitxers csv. Nosaltres hem començat per les de les persones on fem un merge perquè si el node no existeix es creï amb les propietats que especifiquem en el codi, corresponents a les columnes del fitxer de dades. Amb les dades dels habitatges fem el mateix.

A continuació utilitzant les dades del fitxer FAMILIA creem les relacions amb aquest nom i les propietats corresponents. Per fer-ho busquem en els nodes persona aquells que tenen els IDs corresponents amb els de cada fila de l'arxiu de dades.

A més a més creem també les realacions "VIU" entre els nodes persona i els nodes que corresponen a l'habitatge on viuen si es que en tenim registre, hi ha algun node que queda sol perquè no té habitatge o no tenim constància de quin és en les nostres dades.

Per últim creem afegim al nostre graf les relacions "SAME\_AS" que ens indiquen que els dos nodes que uneix representen la mateixa persona en diferents moments, ja que aquestes dades s'han anat recollint en diferents anys i les persones no acostumen a viure sempre en un mateix habitatge.

```
CREATE CONSTRAINT PrimaryKey_Persona IF NOT EXISTS on (p:Persona) ASSERT p.ID IS
UNIQUE;

CREATE CONSTRAINT PrimaryKey_Habitatge IF NOT EXISTS on (h:Habitatge) ASSERT
h.Unique IS UNIQUE;

LOAD CSV WITH HEADERS FROM
'https://docs.google.com/spreadsheets/d/e/2PACX-1vTfU6oJBZhnhzzkV_0-avABPzHTdXy8851
ySDbn2gq32WwaNmYxfiBtCGJGOZsMgCWjzIEGX4Zh1wqe/pub?output=csv' as row

Merge (p:Persona {ID: toInteger(row.Id), Nom: row.name, Cognom: row.surname,
Segon_Cognom: row.second_surname, Any: toInteger(row.Year)})

return NULL;
```

#### LOAD CSV WITH HEADERS FROM

`'https://docs.google.com/spreadsheets/d/e/2PACX-1vT0ZhR6BSO_M72JEmxXKs6GLuOwxm_Oy-0UruLJeX8_R04KAclCuvrwn2OENQhtuvddU5RSJSclHRJf/pub?output=csv' as row`

`Merge (:Habitatge {ID: toInteger(row.Id_Llar), Any: toInteger(row.Any_Padro), Carrer: row.Carrer, Numero: COALESCE(toInteger(row.Numero), 'nan'), Municipi: row.Municipi, Unique: row.Id_Llar + ',' + row.Any_Padro + ',' + row.Municipi})`

`return NULL;`

#### LOAD CSV WITH HEADERS FROM

`'https://docs.google.com/spreadsheets/d/e/2PACX-1vRVOoMAMoxHiGboTjCIHo2yT30CCWgVHgocGnVJxiCTgyurtmqCfAFahHajobVzwXFLwhqajz1fqA8d/pub?output=csv' AS row`

`match (p1:Persona) where toInteger(row.ID_1) = p1.ID`

`match (p2:Persona) where toInteger(row.ID_2) = p2.ID`

`Merge (p1)-[:FAMILIA {Relacio: row.Relacio, Relacio_Harmonitzada: row.Relacio_Harmonitzada}]- (p2)`

`return NULL;`

#### LOAD CSV WITH HEADERS FROM

`'https://docs.google.com/spreadsheets/d/e/2PACX-1vRM4DPeqFmv7w6kLH5msNk6_Hdh1wuExRirgysZKO_Q70L21MKBkDISlyjvdm8shVixl5Tcw_5zCfdg/pub?output=csv' AS row`

`match (p:Persona) where toInteger(row.IND) = p.ID`

`match (h:Habitatge) where toInteger(row.HOUSE_ID) = h.ID and row.Location = h.Municipi and toInteger(row.Year) = h.Any`

`Merge (p)-[:VIU]->(h)`

`return NULL;`

#### LOAD CSV WITH HEADERS FROM

`'https://docs.google.com/spreadsheets/d/e/2PACX-1vTgC8TBmdXhjUOPKJxyiZSpetPYjaRC34gmxHj6H2AWvXTGbg7MLKVdJnwuh5bleer7WLUi0Oigl6wc/pub?output=csv' AS row`

`match (p3:Persona) where toInteger(row.Id_A) = p3.ID`

`match (p4:Persona) where toInteger(row.Id_B) = p4.ID`

`Merge (p3)-[:SAME_AS]- (p4)`

`return NULL;`

```

neo4j$ match (n) detach delete n
neo4j$ CREATE CONSTRAINT PrimaryKey_Persona IF NOT EXISTS on (p:Persona) ASSERT p.ID IS UNIQUE
neo4j$ CREATE CONSTRAINT PrimaryKey_Habitatge IF NOT EXISTS on (h:Habitatge) ASSERT h.Unique IS UNIQUE
neo4j$ LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/e/2PACX-1vTU6oJ827mhz2kV_8-avBPzHTdXy8S1yS0bn2gU32WaaNvXf18COJ02z9MgCjz1EGX4Zhiwq/pub?output=csv" as row Merge (p:Persona {ID: toInteger(row.ID), Nom: row.name_
neo4j$ LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/e/2PACX-1vT8Zt86850_MU73EwxKsG6uOwXn_Oy-BUruL3eX8_B8AKACIcuvrm20ENqhtuvd0JRS35c1H8Jf/pub?output=csv" as row Merge (h:Habitatge {ID: toInteger(row.ID_Llar), Any: to
neo4j$ LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/e/2PACX-1v8V0AMMoxiG6b7JCI6o2y138C8gVigocinVx1C1gyurtmqcFAfAhhaJobvzw7Lwqaz1fQABd/pub?output=csv" AS row match (p1:Persona) where toInteger(row.ID_1) = p1.ID m
neo4j$ LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/e/2PACX-1v8MA0peqfmr7wkiH5ssNk6_Hdh1wufxR1gysZKO_Q78L2IMK8D15iyjvdmshv1x15tCw_5zCFdg/pub?output=csv" AS row match (p:Persona) where toInteger(row.ID) = p.ID matc
neo4j$ LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/e/2PACX-1vTgC8T8mDxhJU0PK3y1Z5petPyJhR34gmxHj0H2ANvXT0bg7MLKVdJmuh5bIeer7MLU180igI6wc/pub?output=csv" AS row match (p3:Persona) where toInteger(row.ID_A) = p3.ID m

```

## Exercici 2

### - Consulta 1

```

1 match (p:Persona)-[:VIU]-(h:Habitatge) where p.Cognom <> 'nan' and h.Any=1866 and h.Municipi = 'CR'
2 return count(*), collect(distinct p.Cognom)

```

count(*)	collect(distinct p.Cognom)
336	["galceran", "ala", "salaf", "rumei", "jallert", "tergallo", "englada", "ala", "reo", "jula", "jallert", "rumei", "parera", "juseu", "canals", "canals", "dalmases", "llop", "llorens", "cald", "astuch", "valde", "astuch", "mparchy", "valde", "llorens", "llorens", "llorens", "casanova", "arnau", "arucas", "capallades"]

En aquest cas, per aquesta consulta, ens hem fixat que encara que a l'enunciat explica que s'ha de fer la query segons els noms dels habitants, als resultats donats es fa la query segons els cognoms. Al codi final hem ficat les dues versions per tal que es puguin contrastar. A la captura es mostra la query amb els cognoms.

### - Consulta 2

```

1 match (h:Habitatge)
2 where h.Municipi='SFL' and h.Any < 1840
3 return h.Any as Any, collect(h.ID) as Identificadors
4 ORDER BY h.Any

```

Any	Identificadors
1833	[95, 96, 101, 103, 105, 107, 109, 111, 94, 96, 97, 98, 100, 104, 102, 106, 110, 113, 115, 116, 119, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192, 194, 196, 198, 200, 202, 204, 206, 208, 210, 212, 214, 216, 218, 220, 222, 224, 226, 228, 230, 232, 234, 236, 238, 240, 242, 244, 246, 248, 250, 252, 254, 256, 258, 260, 262, 264, 266, 268, 270, 272, 274, 276, 278, 280, 282, 284, 286, 288, 290, 292, 294, 296, 298, 300, 302, 304, 306, 308, 310, 312, 314, 316, 318, 320, 322, 324, 326, 328, 330, 332, 334, 336, 338, 340, 342, 344, 346, 348, 350, 352, 354, 356, 358, 360, 362, 364, 366, 368, 370, 372, 374, 376, 378, 380, 382, 384, 386, 388, 390, 392, 394, 396, 398, 400, 402, 404, 406, 408, 410, 412, 414, 416, 418, 420, 422, 424, 426, 428, 430, 432, 434, 436, 438, 440, 442, 444, 446, 448, 450, 452, 454, 456, 458, 460, 462, 464, 466, 468, 470, 472, 474, 476, 478, 480, 482, 484, 486, 488, 490, 492, 494, 496, 498, 500, 502, 504, 506, 508, 510, 512, 514, 516, 518, 520, 522, 524, 526, 528, 530, 532, 534, 536, 538, 540, 542, 544, 546, 548, 550, 552, 554, 556, 558, 560, 562, 564, 566, 568, 570, 572, 574, 576, 578, 580, 582, 584, 586, 588, 590, 592, 594, 596, 598, 600, 602, 604, 606, 608, 610, 612, 614, 616, 618, 620, 622, 624, 626, 628, 630, 632, 634, 636, 638, 640, 642, 644, 646, 648, 650, 652, 654, 656, 658, 660, 662, 664, 666, 668, 670, 672, 674, 676, 678, 680, 682, 684, 686, 688, 690, 692, 694, 696, 698, 700, 702, 704, 706, 708, 710, 712, 714, 716, 718, 720, 722, 724, 726, 728, 730, 732, 734, 736, 738, 740, 742, 744, 746, 748, 750, 752, 754, 756, 758, 760, 762, 764, 766, 768, 770, 772, 774, 776, 778, 780, 782, 784, 786, 788, 790, 792, 794, 796, 798, 800, 802, 804, 806, 808, 810, 812, 814, 816, 818, 820, 822, 824, 826, 828, 830, 832, 834, 836, 838, 840, 842, 844, 846, 848, 850, 852, 854, 856, 858, 860, 862, 864, 866, 868, 870, 872, 874, 876, 878, 880, 882, 884, 886, 888, 890, 892, 894, 896, 898, 900, 902, 904, 906, 908, 910, 912, 914, 916, 918, 920, 922, 924, 926, 928, 930, 932, 934, 936, 938, 940, 942, 944, 946, 948, 950, 952, 954, 956, 958, 960, 962, 964, 966, 968, 970, 972, 974, 976, 978, 980, 982, 984, 986, 988, 990, 992, 994, 996, 998, 1000]
1838	[321, 324, 326, 328, 330, 332, 334, 336, 338, 340, 342, 344, 346, 348, 350, 352, 354, 356, 358, 360, 362, 364, 366, 368, 370, 372, 374, 376, 378, 380, 382, 384, 386, 388, 390, 392, 394, 396, 398, 400, 402, 404, 406, 408, 410, 412, 414, 416, 418, 420, 422, 424, 426, 428, 430, 432, 434, 436, 438, 440, 442, 444, 446, 448, 450, 452, 454, 456, 458, 460, 462, 464, 466, 468, 470, 472, 474, 476, 478, 480, 482, 484, 486, 488, 490, 492, 494, 496, 498, 500, 502, 504, 506, 508, 510, 512, 514, 516, 518, 520, 522, 524, 526, 528, 530, 532, 534, 536, 538, 540, 542, 544, 546, 548, 550, 552, 554, 556, 558, 560, 562, 564, 566, 568, 570, 572, 574, 576, 578, 580, 582, 584, 586, 588, 590, 592, 594, 596, 598, 600, 602, 604, 606, 608, 610, 612, 614, 616, 618, 620, 622, 624, 626, 628, 630, 632, 634, 636, 638, 640, 642, 644, 646, 648, 650, 652, 654, 656, 658, 660, 662, 664, 666, 668, 670, 672, 674, 676, 678, 680, 682, 684, 686, 688, 690, 692, 694, 696, 698, 700, 702, 704, 706, 708, 710, 712, 714, 716, 718, 720, 722, 724, 726, 728, 730, 732, 734, 736, 738, 740, 742, 744, 746, 748, 750, 752, 754, 756, 758, 760, 762, 764, 766, 768, 770, 772, 774, 776, 778, 780, 782, 784, 786, 788, 790, 792, 794, 796, 798, 800, 802, 804, 806, 808, 810, 812, 814, 816, 818, 820, 822, 824, 826, 828, 830, 832, 834, 836, 838, 840, 842, 844, 846, 848, 850, 852, 854, 856, 858, 860, 862, 864, 866, 868, 870, 872, 874, 876, 878, 880, 882, 884, 886, 888, 890, 892, 894, 896, 898, 900, 902, 904, 906, 908, 910, 912, 914, 916, 918, 920, 922, 924, 926, 928, 930, 932, 934, 936, 938, 940, 942, 944, 946, 948, 950, 952, 954, 956, 958, 960, 962, 964, 966, 968, 970, 972, 974, 976, 978, 980, 982, 984, 986, 988, 990, 992, 994, 996, 998, 1000]
1839	[721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000]

### - Consulta 3

```

1 match (p:Persona)-[:VIU]-(h:Habitatge)-[:VIU]-(p2:Persona)
2 where toLower(p.Nom)='rafel' and toLower(p.Cognom)='martí' and h.Any=1838
3 return p.Nom, collect(p2.Nom)

```

p.Nom	collect(p2.Nom)
"rafel"	["felipe", "salvadora", "franco", "maria", "jha", "miquel", "jha", "jh"]

#### - Consulta 4

```
1 match (p:Persona)-[:SAME_AS]->(n) where toLower(p.Nom)='miguel' and toLower(p.Cognom)='ballester'
2 return n
```



#### - Consulta 5

```
1 match(p:Persona)-[:SAME_AS]->(n)
2 where p.Nom = 'antonio' and p.Cognom = 'farran'
3 return n.Nom as Nom,n.Cognom as Cognom1,n.Segon_Cognom as Cognom2,"SAME_AS" as Tipus
4 UNION ALL
5 match(p:Persona)-[:FAMILIA]->(n)
6 where p.Nom = 'antonio' and p.Cognom = 'farran'
7 return n.Nom as Nom,n.Cognom as Cognom1,n.Segon_Cognom as Cognom2,"FAMILIA" as Tipus
```

Nom	Cognom1	Cognom2	Tipus
"antonio"	"farran"	"sola"	"SAME_AS"
"antonio"	"farran"	"sola"	"SAME_AS"
"separanca"	"farran"	"sola"	"FAMILIA"
"separanca"	"sola"	"gessen"	"FAMILIA"
"catalina"	"farran"	"sola"	"FAMILIA"
"sandro"	"farran"	"sola"	"FAMILIA"
"sola"	"farran"	"sola"	"FAMILIA"
"antonio"	"farran"	"sola"	"FAMILIA"

#### - Consulta 6

```
1 match (:Persona)-[x:FAMILIA]-(:Persona)
2 where x.Relacio_Harmonitzada <> 'null'
3 return distinct x.Relacio_Harmonitzada as Relacions_Familiars
```

Relacions_Familiars
"separanca"
"sola"
"sola"
"sola"
"sola"
"sola"
"sola"
"sola"
"sola"
"sola"

## - Consulta 7

```
1 match (h:Habitatge[Municipi:'SFL'])
2 where h.Carrer <> 'nan' and h.Numero <> 'nan'
3 return (h.Carrer) as Carrer, h.Numero as Numero_Carrer, count(distinct h.Any) as Total, collect(distinct h.Any) as Anys, collect(h.ID) as Ids
4 order by Total desc, h.Carrer, h.Numero limit 10
```

Carrer	Numero_Carrer	Total	Anys	Ids
"Cros"	9	5	[1833, 1839, 1878, 1881, 1889]	[150, 610, 467, 367, 347]
"Cros"	16	5	[1833, 1839, 1878, 1881, 1889]	[150, 616, 473, 372, 368]
"Cros"	18	5	[1833, 1839, 1878, 1881, 1889]	[160, 618, 475, 374, 369]
"Cros"	23	5	[1833, 1839, 1878, 1881, 1889]	[166, 624, 488, 388, 373]
"gheda"	1	5	[1833, 1839, 1878, 1881, 1889]	[84, 721, 514, 509, 483]
"gheda"	2	5	[1833, 1839, 1878, 1881, 1889]	[86, 722, 515, 510, 488]
"gheda"	3	5	[1833, 1839, 1878, 1881, 1889]	[87, 723, 514, 516, 511, 484]
"gheda"	4	5	[1833, 1839, 1878, 1881, 1889]	[88, 725, 517, 512, 489]
"gheda"	8	5	[1833, 1839, 1878, 1881, 1889]	[102, 738, 528, 516, 508]
"gheda"	9	5	[1833, 1839, 1878, 1881, 1889]	[104, 739, 530, 521, 517, 487]

## - Consulta 8

```
1 match(h:Habitatge)-[:VIU]-(p:Persona)-[r:FAMILIA]->(n), (h:Habitatge)-[:VIU]-(pe:Persona)-[re:FAMILIA]->(n)
2 where h.Municipi='ca' and (tolower(r.Relacio)='cap' or tolower(r.Relacio)='cabeza') and (tolower(re.Relacio)='hija' or tolower(re.Relacio)='hija' or tolower(re.Relacio_Harmonitzada) = 'fill' or tolower(re.Relacio_Harmonitzada) = 'fills')
3 return distinct p.Num as nombre, p.Cognom as apellido, p.Segon_Cognom as segundo_apellido, size(collect(pe)) as fills
4 order by fills DESC limit 20
```

nombre	apellido	segundo_apellido	fills
"judo"	"wibsch"	"judo"	7
"judo"	"bargallo"	"bargallo"	6
"jose"	"ula"	"domenach"	6
"jose"	"canals"	"ula"	6
"jose"	"canals"	"ula"	6
"bardo"	"jaldres"	"jarrera"	6
"bancroft"	"aragay"	"igol"	5
"judo"	"bargallo"	"arriagap"	5
"jose"	"jerry"	"bargallo"	5
"jose"	"tobals"	"ula"	5

## - Consulta 9

```
1 match(h1:Habitatge) where h1.Any = 1881 and h1.Municipi = 'SFL' with count(distinct(h1)) as num
2 match(h:Habitatge)-[:VIU]-(p:Persona)-[r:FAMILIA]->(n)
3 where h.Any = 1881 and h.Municipi = 'SFL' and (r.Relacio_Harmonitzada) STARTS WITH 'fill'
4 return num as habitatges, count(distinct(p)) as fills, toFloat(count(distinct(p)))/toFloat(num) as mitjana
```

habitatges	fills	mitjana
596	1292	2.1677852348893287

## - Consulta 10

```
1 match (h:Habitatge)-[:VIU]-(p:Persona)
2 where h.Municipi='SFL'
3 with h.Any as Any, h.Carrer as Carrer, count(*) as num_hab
4 Order by Any, num_hab
5 return Any, collect(Carrer)[0], collect(num_hab)[0]
```

Any	collect("carro")[0]	collect("sum_hab")[0]
1833	"carroza de la part de molinos de rey"	5
1838	"carrotera de bamba"	30
1839	"casas del Zorramy"	3
1878	"carretera"	2
1981	"Carretera"	6
1989	"a n antorio"	1

### Exercici 3

#### a) Estudi de les components connexes i la seva estructura en funció de la mida

**Primera motivació:** veure l'evolució de les estructura familiars al llarg dels anys.

**Primer creem un graf temporal per cada any:**

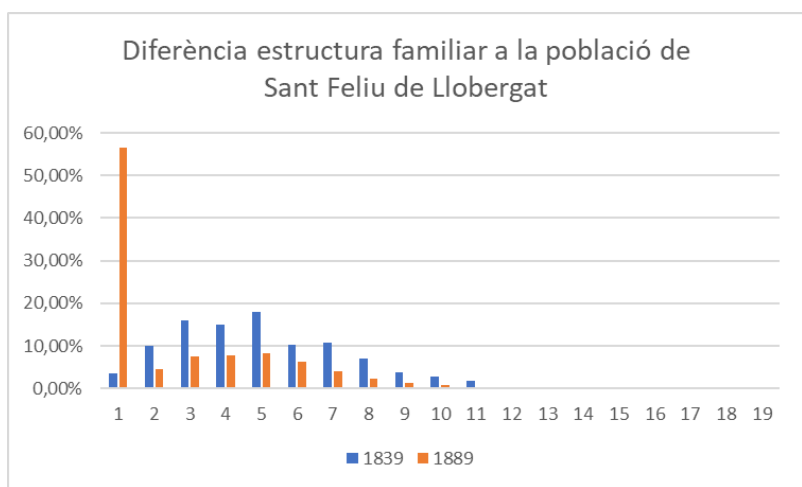
```
CALL gds.graph.create.cypher('relacions1881', 'MATCH (p:Persona) where p.Any = 1881 return id(p) as id', 'MATCH (un:Persona)-[f:FAMILIA]->(dos:Persona) return id(un) as source, id(dos) as target', {validateRelationships:False})
```

```
YIELD graphName AS graph, nodeQuery, nodeCount AS nodes, relationshipQuery, relationshipCount AS rels
```

**A posteriori fem servir gds.wcc.stream per trobar totes les famílies que hi ha i la seva mida:**

```
call gds.wcc.stream('relacions')
yield componentId , nodeId
with componentId , collect(nodeId) as nodes , size(collect(nodeId)) as mida
return distinct mida, size(collect(componentId)) as total_families
order by mida
```

TOTAL DE MEMBRES FAMILIARS	Any 1839	Any 1889	Any 1839	Any 1839
1	13	638	3,42%	56,41%
2	38	51	10,00%	4,51%
3	61	85	16,05%	7,52%
4	57	87	15,00%	7,69%
5	68	93	17,89%	8,22%
6	39	72	10,26%	6,37%
7	41	45	10,79%	3,98%
8	27	26	7,11%	2,30%
9	14	15	3,68%	1,33%
10	11	9	2,89%	0,80%
11	7	4	1,84%	0,35%
12	1	2	0,26%	0,18%
13	1	2	0,26%	0,18%
14	0	0	0,00%	0,00%
15	1	0	0,26%	0,00%
16	1	1	0,26%	0,09%
17	0	0	0,00%	0,00%
18	0	0	0,00%	0,00%
19	0	1	0,00%	0,09%



Com podem comprovar en la gràfica hi ha una evidència clara d'estructures familiars diferents al municipi de Sant Feliu entre 1839 i 1889. Ambdues presenten una distribució de número d'habitants semblant, però destaca que l'any 1889 un 50% de la població de Sant Feliu no hi té família, fet que a l'any 1839 no arribava ni al 5%. Aquest fet suposem



que és per l'arribada de nouvinguts a Sant Feliu, els quals arribaven a la ciutat sols i sense família, segurament en busca de feina.

## **b) Semblança entre els nodes**

### **Creació de les arestes “MATEIX\_HAB”:**

```
MATCH (h1:Habitatge), (h2:Habitatge)
WHERE h1.Numero <> 'nan' AND h2.Numero <> 'nan' AND h1.Municipi <> 'null' AND
h2.Municipi <> 'null' AND h1.Municipi=h2.Municipi AND h1.Carrer=h2.Carrer AND
h1.Numero=h2.Numero AND h1.Any>h2.Any
MERGE (h1)-[:MATEIX_HAB]->(h2)
RETURN h1, h2
```

Noves arestes després d'aquesta query: +3504 arestes

### **Creació del graf:**

```
CALL gds.graph.create('Similar', ['Persona','Habitatge'],['VIU', 'FAMILIA',
'MATEIX_HAB'])
```

### **Càlcul de la similaritat:**

```
CALL gds.nodeSimilarity.write('Similar',{ writeRelationshipType:'SIMILAR',
writeProperty: 'score', similarityCutoff:0.45, topK:5 })
YIELD nodesCompared, relationshipsWritten
```

Aquesta query ens diu que s'han comparat 15864 nodes i s'han escrit 63412 relacions de similaritat, per tant de mitjana un node es considera similar a 4 altres.

D'aquests nodes que hem unit amb la relació SIMILAR, comprovem quins tenen també entre ells una aresta de tipus SAME\_AS per així veure quants dels nodes que representen una mateixa persona estem detectant com a similars. Ho fem amb aquesta query:

```
MATCH (p1:Persona)-[:SAME_AS]-(p2:Persona)
WHERE (p1)-[:SIMILAR]-(p2)
RETURN p1, p2
```

L'output obtingut ens diu que no ha trobat cap cas en que es compleixi aquesta condició. Veient aquesta resposta hem provat de redefinir les relacions de similaritat canviant el valor de similarityCutoff, així com el de topK però els resultats han estat els mateixos pels diferents valors provats. Això per al cas dels nodes de tipus Persona, per comprovar les similaritats trobades en el cas dels nodes de tipus Habitatge que prèviament hem unit amb una aresta de tipus “MATEIX\_HAB” fem servir aquesta query:

```

MATCH (p1:Habitatge)-[:MATEIX_HAB]-(p2:Habitatge)
WHERE (p1)-[:SIMILAR]-(p2)
RETURN count(*) as nCoincidencies

```

Retorna 750 coincidències, això vol dir que ha detectat 750 de les 3504 relacions en les que dos habitatges son iguals, per tant té un 21.4% d'encert. Ha generat un total de 1576 relacions de similar entre els habitatges. Tenint en compte això vol dir que de les relacions generades un 47,6% eren correctes.

Això implica també que hi ha molts nodes que ha detectat com a similars molts nodes persona que nosaltres considerem que no ho son. Recordem que la consulta creava 63.412 relacions, per tant creava 61.836 (63.412-1.576) relacions similars entre persones (o entre persones i habitatges), de les quals cap és correcta. Per tant podem dir que aquest algorisme no te la precisió que volíem.

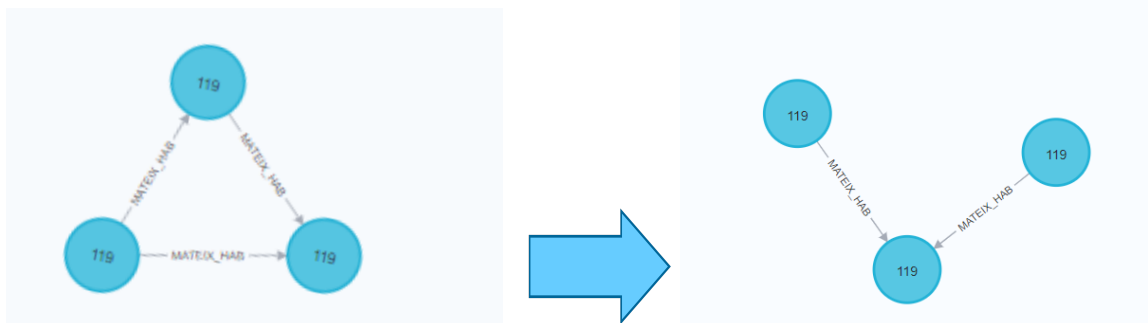
Ara provem executant després de la generació de mateixos habitatges aquesta consulta però repetint tot el procés de la mateixa manera:

```

MATCH (h1)-[:MATEIX_HAB]-(h2)-[:MATEIX_HAB]-(h3)
DELETE r

```

Aquesta consulta fa que els mateixos habitatges apuntin tots a un de sol, el que tingui el menor any. Per tant en comptes d'un clúster relativament comunicat es pot veure com un ventall on tots apunten al mateix.



Resultats:

```

MATCH (p1:Persona)-[:SAME_AS]-(p2:Persona)
WHERE (p1)-[:SIMILAR]-(p2)
RETURN p1, p2

```

No retorna res de la mateixa manera que en l'altra prova

```

MATCH (p1:Habitatge)-[:MATEIX_HAB]-(p2:Habitatge)
WHERE (p1)-[:SIMILAR]-(p2)
RETURN count(*) as nCoincidencies

```

Retorna 0 coincidències i per tant falla completament. En aquest cas la consulta havia generat 1190 relacions similars entre habitatges i 57636 relacions similars entre persones.

Per tant aquesta prova el resultat és pitjor i per aquest motiu ens quedarem amb els resultats originals.

Després de fer aquest anàlisi podem veure que és coherent que els resultats no siguin bons i no ens ajudin a l'hora de buscar similitud entre nodes que representen la mateixa persona en diferents anys. Això es deu a que la node Similarity utilitzada es basa en buscar nodes connectats seguint uns patrons específics i les persones no tenen perquè formar part de nuclis familiars amb la mateixa estructura durant tota la seva vida. És a dir, fins a certa edat si que segurament les persones viuen amb els mateixos familiars, normalment pares i germans si en tenen o fins i tot padrins, això no canvia cada any, però acostuma arribar un moment en que les persones s'independitzen i formen altres nuclis familiars que no tenen perquè seguir el mateix patró que el nucli del que havien format part prèviament.

A més a més, que un mateix habitatge pugui estar representat per nodes diferents en funció de l'any també dificulta la búsqueda de patrons, si els nodes habitatge fossin constants al llarg dels anys segurament hauríem obtingut resultats millors.

## **Treball en equip**

Per fer aquest treball hem parlat entre tots els membres del grup com volíem fer cada apartat i quina era la millor manera de resoldre els diferents exercicis.

Un cop vam tenir clar com desenvolupar cada tasca ens les vam repartir per centrar-nos cadascú en una part. En Daniel es va encarregar més de la part de carregar les dades i va fer alguna aportació en la part de les consultes, d'aquesta se'n va encarregar majoritàriament en Sergio. Finalment l'exercici 3 s'ha repartit entre el Xavier i l'Abril, un apartat cadascú. Tot i que ens haguem centrat cadascú en una part sempre que a algú li ha sorgit un dubte l'hem comentat i resolt entre tots, aportant el que sabíem.

En última instància, per fer l'informe cadascú ha escrit més sobre la seva part ja que es la que teníem més present i més clara. però tots ho hem revisat i hem afegit el que hem considerat que calia esmentar si encara no hi constava.