## DATATHON FME

### AED CHALLENGE FIND THE PERFECT TEAM

Adriana Aguiló Amal Dokkar Jordina Gavaldà Alèxia Escudero



### PARÀMETRES QUE HEM TINGUT EN COMPTE

id name email age year\_of\_study shirt size university dietary\_restrictions programming\_skills experience\_level hackathons done interests

preferred\_role objective interest\_in\_challenges preferred\_languages friend\_registration preferred\_team\_size availability introduction technical\_project future excitement fun\_fact

#### **Multipliers:**



#### **OBJECTIVE**

Hem definit un diccionari amb paraules clau que hem tret d'alguns textos i hem comptat el nombre de cops que apareixien en la resta.

D'aquesta manera ens ha quedat un vector amb tantes coordenades com categories a classificar. La proximitat entre dos objectius és el producte escalar entre els dos vectors.

#### **PROGRAMMING SKILLS**

Hem processat la llista de programming skills per tal que sigui un vector de enters amb un nombre describint la skill a cada posició.

Dues persones són més compatibles quant majors skills tingui el màxim element a element dels seus dos vectors

### NO HI HA UNA ÚNICA SOLUCIÓ VÀLIDA!

Fem una mitjana pondera amb uns multiplicadors definits per l'usuari.

# JSUARI 1: SONÈIXER GENT

++ interesos similars

+ amb amics

- + curs similar
- habilitats de programació
- -- rol desitjat

# USUARI 2: COMPETITIVITAT

++ habilitats de programació

- + rol desitjat
- + disponibilitat
- cursos similars
- -- amb amics

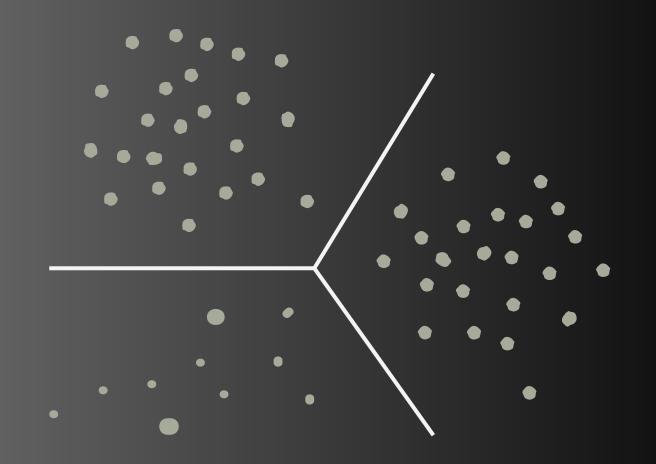
## K-MEANS CONSTRAINED

- No té en compte el tamany d'equip preferent
- No considera la transitivitat de les preferències
- k Means clustering algorithm adaptat
- Limitem 3-4 persones per cluster
- Repositori de la llibreria:
   https://github.com/joshlk/
   k-means-constrained

### BEAM SEARCH

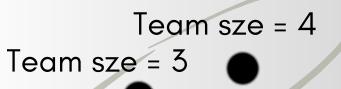
- Té en compte les preferències de tamany d'equip
- Considera una fusió de les característiques quan les persones s'uneixen a un equip
- Aproximació greedy al clustering ideal escollint cada cop les W millors opcions

# K-MEANS CONSTRAINED

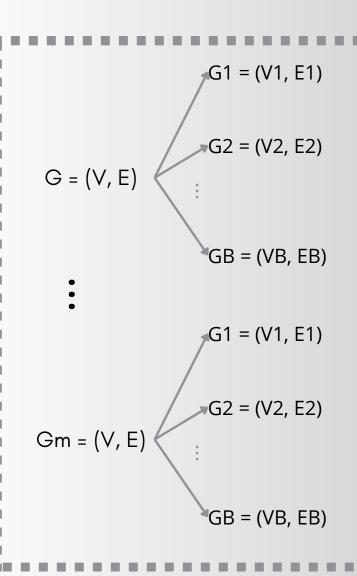


### BEAM SEARCH

Team sze = 3



Provem de juntar les *B* parelles més properes



Ens quedem amb les *M* millor opcions



$$G1 = (V1, E1)$$

Gm = (Vm, Em)

I repetim!

### e3484b47-0c0a-41d7-9750-a21adb889205

a00428a2-14a2-4fb1-838a-22394778ec88 662230f0-a069-402e-8270-5e7bce63718c 57fffc51-bccs-4c25-afbe-155255e4all/9 50ac00ab-42d5-40ae-a7da-ad69a687b7cb

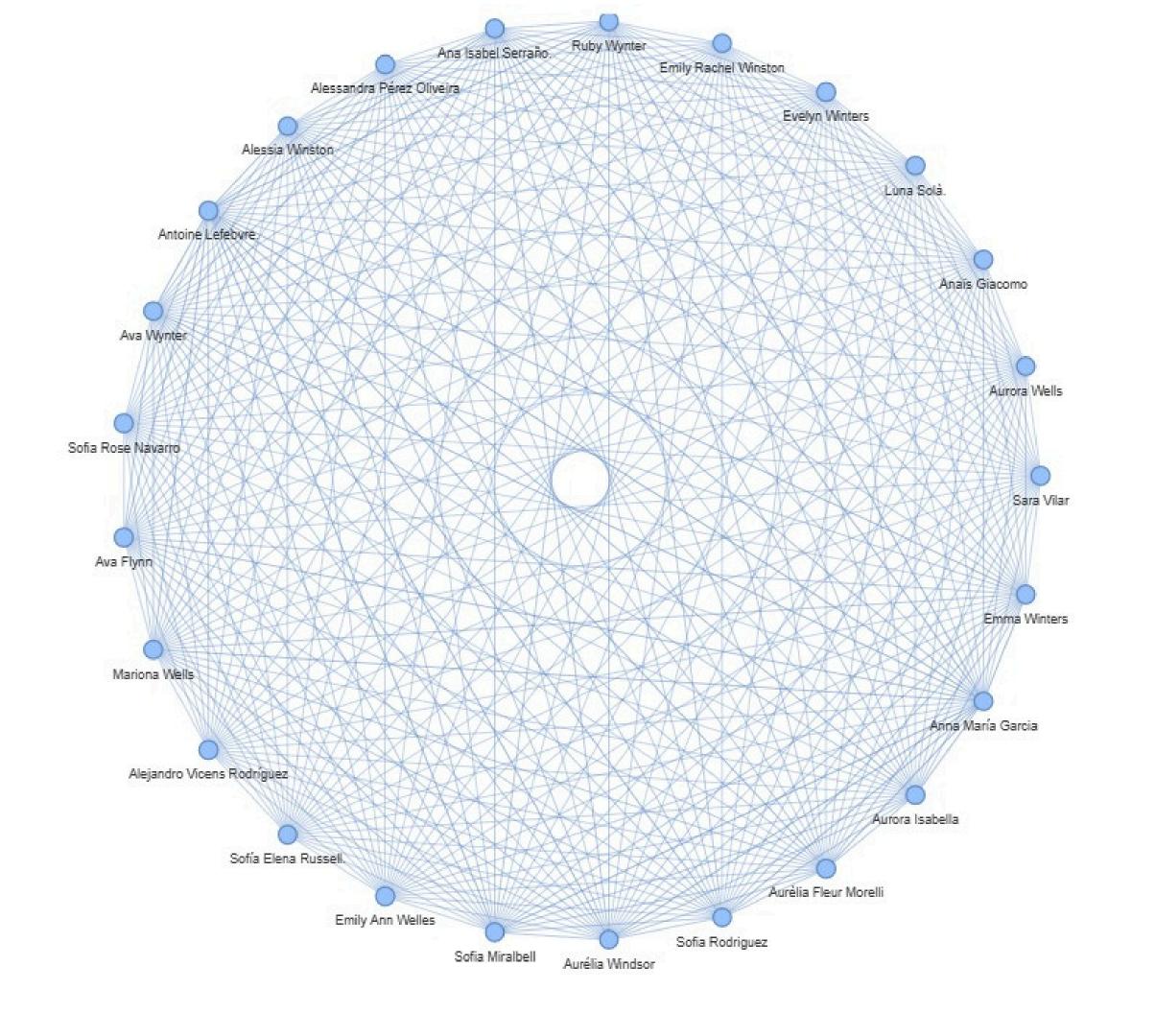
1984f7983-70c7-49e9-afe5-7b6fdf59db5b d1c120a3-498a-4a25-89287#823431#0ed-4204-9ef3-31b2dc51615d

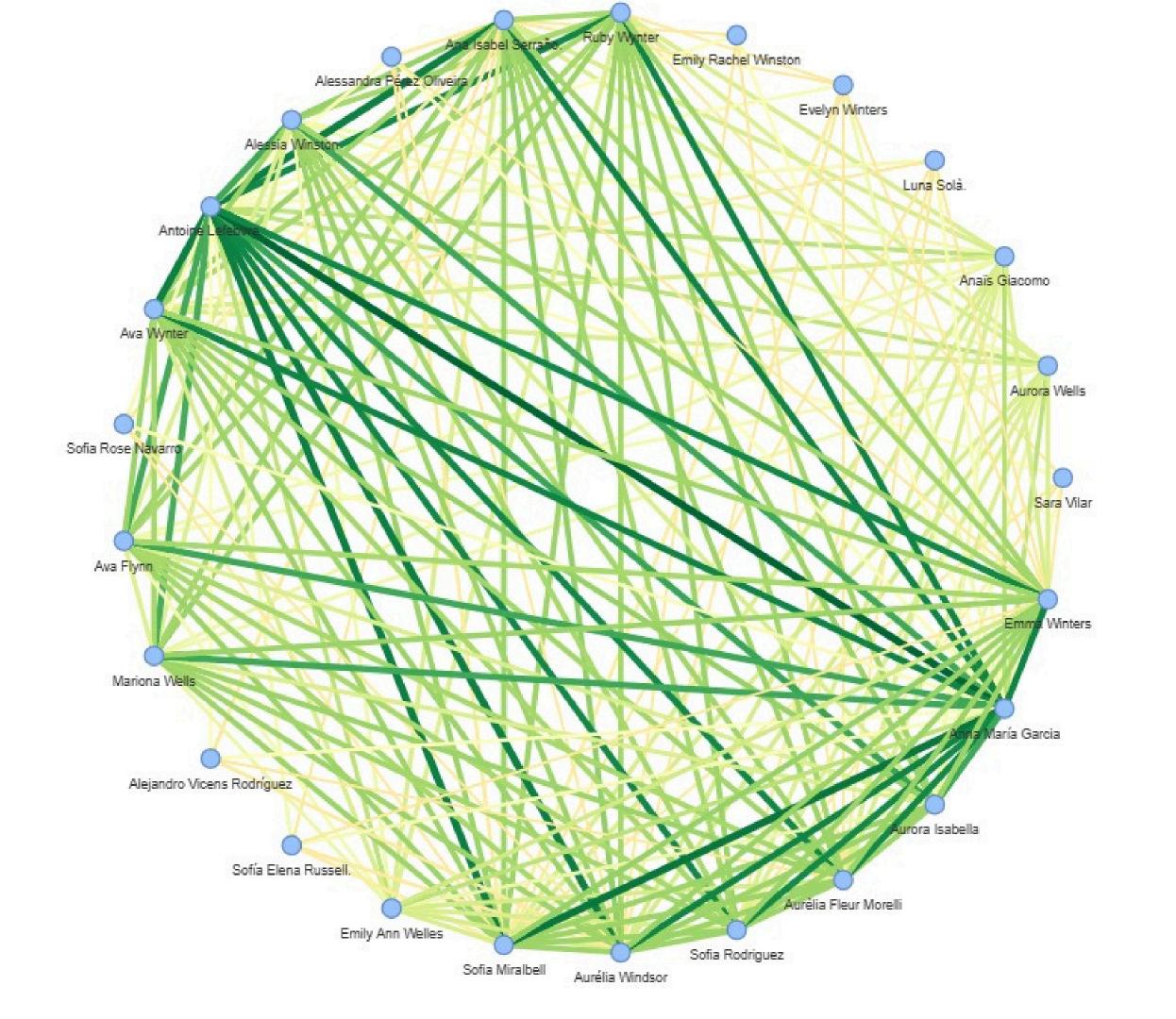
81072249-9652-477e-9efU-208csace28e8 824f4877-d31a-4egf-a7de-5u77392fg1aea7-5eDe-4fao-5919-cdddd1c11d14

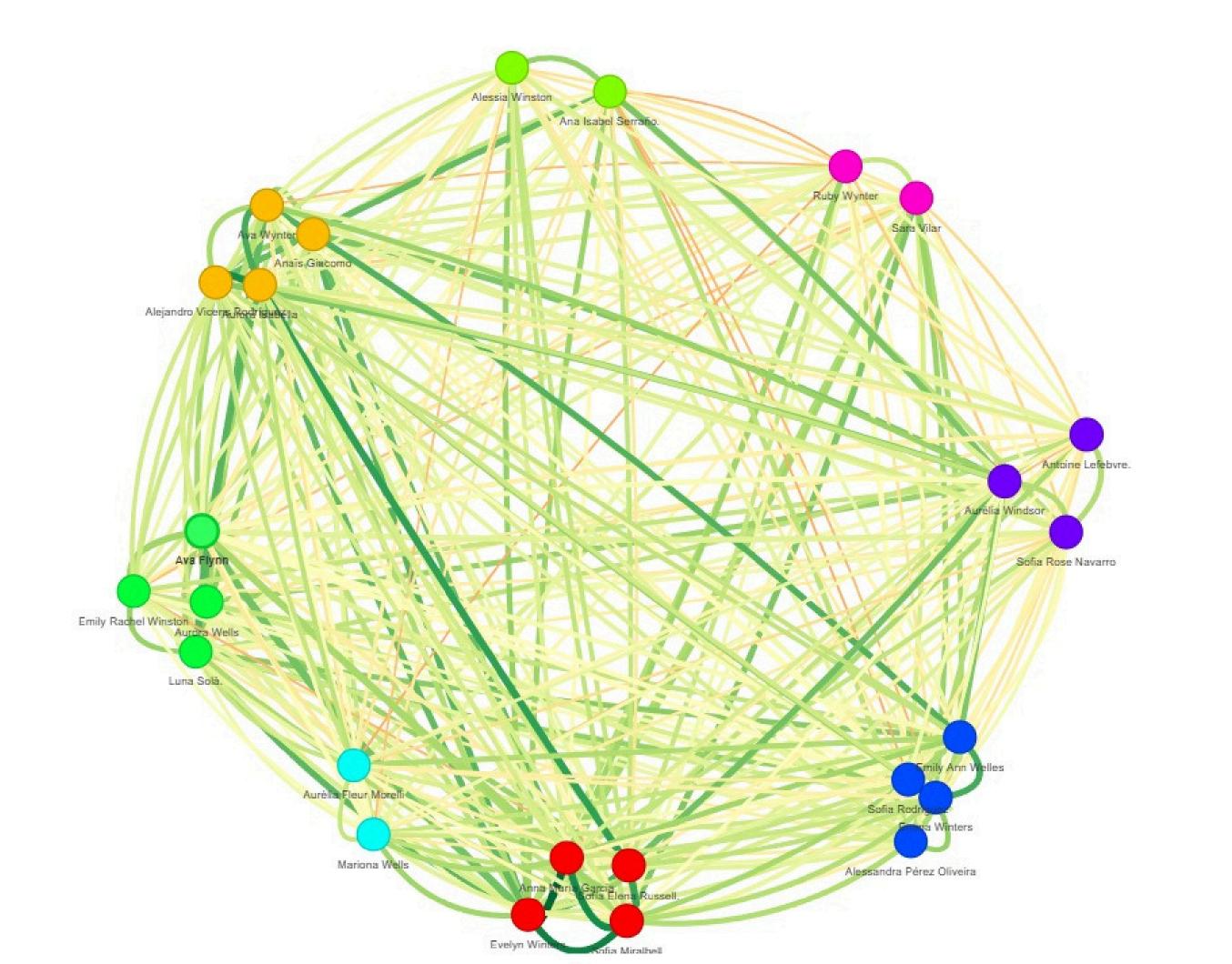
2647ff5e-eb6b-4kf6**ce57 0 08cc543 (passentility 035e**f3**co**\$-9c71-520a23c808a0

55ef7 po8-be39-4572-912d-67 mp360aae4 93cce408-4699 \*\*53-bc4d-bc5dd6281e3c

8d317b45-ad3e-42c3-996a-fc566a0d6068





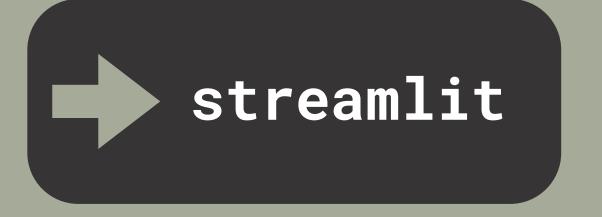


1. DEFINIR ELS
MULTIPLICADORS
(usuari a través de la
interfície de streamlit)

3. CALCULAR ELS GRUPS
QUE MINIMITZIN EL PES
TOTAL
(i renderitzar-ho)

NOTA: Abans s'ha preprocessat el JSON, canviant cursos a nombres, skills a vectors, etc.

2. CREAR EL GRAF AMB
PESOS (segons els criteris
que hem explicat)



## "D'ACORD, PERÒ EL JSON TÉ 924 ENTRADES, I NOMÉS ENS HEU MOSTRAT EXEMPLES PETITS..."

Ho hem fet amb totes les dades. En un notebook.

Que ha estat bastaaaaant temps executant-se però a qui li importa ahahah

#### GRÀCIES PER LA VOSTRA ATENCIÓ