



UAB
Universitat Autònoma de Barcelona

Grau d'Estadística Aplicada

Anàlisi de dades mediambientals mitjançant teoria de xarxes

Alumna: Esther Amores Gago
NIU: 1497189

Tutor: Joan Gasull Jolis



Barcelona
Juliol 2021

Vull mostrar el meu agraïment a les persones que m'han ajudat a desenvolupar aquest treball, ja sigui tant pel que fa a l'àmbit acadèmic com el personal. En primer lloc, vull donar les gràcies al meu tutor, en Joan, pel seu compromís, els seus consells a l'hora d'elaborar el treball i, sobretot, per donar-me llibertat en quant a l'enfoc i desenvolupament d'aquest. També vull agrair a la meva mare i a la meva germana el seu suport incondicional, així com també a la meva parella i el cercle més proper d'amistats, doncs m'han animat i ajudat quan més falta em feia.

Resum

En el present treball s'introdueix la teoria de grafs, una eina molt potent que es basa en l'estudi de xarxes per modelitzar relacions entre objectes. Mitjançant les dades extretes d'Eurostat sobre l'enviament de residus transfronterers, s'hi aplica l'anàlisi descriptiva de xarxes, és a dir: la visualització i la caracterització numèrica de la xarxa d'importacions de l'any 2018, que és el darrer any que es contempla en la base de dades. S'ha creat una eina mitjançant la llibreria **Shiny** que permet representar de forma dinàmica les interaccions entre els països, així com proporcionar informació resumida dels enviaments de residus. Els països que tenen un grau en els nodes més elevat es relacionen amb el que tenen graus més baixos, i no s'han trobat diferències entre les xarxes de residus perillosos en relació les xarxes de residus no perillosos.

Resumen

En el presente trabajo se introduce la teoría de grafos, una herramienta muy potente que se basa en el estudio de redes para modelizar relaciones entre objetos. Mediante los datos extraídos de Eurostat sobre el envío de residuos transfronterizos de desechos, se aplica el análisis de redes, es decir: la visualización y la caracterización numérica de la red de importaciones del año 2018, que es el último año que se contempla en la base de datos. Se ha creado una herramienta mediante la librería **Shiny** que permite representar de forma dinámica las interacciones entre los países, así como también proporcionar información resumida de los envíos de residuos. Los países que tienen un mayor grado en los nodos se relacionan con los que tienen grados más bajos, y no se han encontrado diferencias entre las redes de residuos peligrosos en relación las redes de residuos no peligrosos.

Abstract

Network analysis, which is a very powerful tool that is based on the study of networks to model relationships between objects, is introduced in this paper. Using the transboundary shipments of waste dataset extracted from Eurostat, the network analysis is applied, that is to say: the visualization and the numerical characterization of the import network for the year 2018, which is the last year contemplated in the database. A tool has been created using **Shiny** package, and it allows the interactions between countries to be dynamically represented, as well as to provide summary information on waste shipments. The countries that have a higher degree in nodes are related to those with lower grades, and no differences have been found between hazardous waste networks in relation to non-hazardous waste networks.

Índex

1	Introducció	5
1.1	Motivació	5
1.2	Objectius	5
2	Marc del treball	7
2.1	Les xarxes	7
2.2	Tipus d'anàlisi de xarxes	7
2.2.1	Visualització i caracterització de dades	8
2.2.2	Modelització i inferència de xarxes	8
2.3	Teoria de grafs	9
2.3.1	Fonaments	9
2.3.2	Components connectats	11
2.3.3	Distàncies i camins	11
2.3.4	Teoria algebraica de grafs	12
2.4	Tractament de les dades	12
2.4.1	Informació sobre les dades	12
2.4.2	La base de dades	14
2.4.3	Preprocessament de les dades	15
2.4.4	Selecció de variables	16
3	Anàlisi de xarxes	17
3.1	Introducció	17
3.2	Visualització	18
3.2.1	igraph	18
3.2.2	plotly i visNetwork	19
3.2.3	Shiny	20
3.3	Anàlisi descriptiva	22
3.3.1	Característiques dels nodes i els enllaços	22
3.3.2	Cohesió de la xarxa	26
3.3.3	Connectivitat, talls i fluxos	30
3.3.4	Segmentació de grafs	31
3.3.5	Agrupació jeràrquica	31
3.3.6	Assortativitat	33
3.4	Anàlisi descriptiva II	34
4	Conclusions	43
5	Referències	45

Annexos

A	Base de dades	ii
B	Preprocessament de les dades	v
C	Taules	xiv
D	Anàlisis descriptives	xx
E	App web Shiny	xxxviii
F	Informació de la sessió	lii

1 | Introducció

1.1 Motivació

En els darrers anys el canvi climàtic ha estat en el punt de mira dels diversos àmbits de la societat, ja sigui mitjançant els moviments socials de l'ecologisme com del desenvolupament de noves tecnologies que podrien frenar aquest repte global. Això últim implica una relació directa amb l'economia dels països, així com també de la presa de decisions polítiques d'aquests. L'estadística pot resultar una eina molt útil en aquests moments gràcies a l'era de la informació en què ens trobem, que de fet es tracta de l'era de la *Big Data*: es generen i s'emmagatzemen dades de forma massiva que, si es tracten de la manera adequada, poden proporcionar informació que no es pot percebre a primera vista. Així doncs, existeixen moltes entrades a internet que donen accés a aquestes dades, l'Eurostat, l'oficina d'estadística de la Unió Europea (*Who we are*), n'és una d'elles.

S'hi poden trobar estadístiques classificades per temàtiques diferents, en tenen una específica per al medi ambient i l'energia, respectivament, i fins i tot per al canvi climàtic, el qual defineixen com a un canvi en els patrons climàtics degut a les activitats humanes, que va més enllà de la variabilitat natural del clima. A més a més, remarquen la importància de disposar d'informació d'alta qualitat per tal de controlar el progrés en la reducció de les emissions de gasos d'efecte hivernacle, que és una de les causes més destacades, i també per a analitzar els motors, els impactes i l'adaptació al canvi climàtic (*Climate change – Overview*).

Els gasos d'efecte hivernacle són una de les causes del canvi climàtic, i alguns d'aquests en són el diòxid de carboni, el metà, els compostos halogenats, l'ozó troposfèric o l'òxid de nitrogen. Aquests es produeixen per activitats com el crematori de combustibles fòssils per la generació d'electricitat, el transport, la indústria i l'edificació. Però també són provocats per la ramaderia, l'agricultura, el tractament d'aigües residuals i els abocadors, entre d'altres *¿Qué es el cambio climático?*

Es podria investigar sobre molts aspectes diferents que engloben el canvi climàtic, que no són pocs, però la gestió dels residus és un tema prou desconegut actualment i, si més no, la societat no tendeix a centrar-se en elles fins que no suposen un problema per a la població, com està passant a hores d'ara.


1.2 Objectius

L'objectiu principal d'aquest treball és veure mitjançant l'anàlisi de xarxes quines relacions hi ha entre les importacions de residus d'Estats Membres de la Unió Europea (UE) i agregats de la UE cap a països socis (Estats Membres de la UE, entre d'altres).

A més a més, també es pretén avaluar quines diferències hi ha entre els residus classificats com a perillosos i no perillosos segons els països. Concretament, es treballarà sobre el tipus d'anàlisi de xarxes que tracta sobre la visualització i la caracterització de dades, fent èmfasi en la representació dels grafs mitjançant la creació d'una app web Shiny. Per tant, un altre propòsit és aprendre el funcionament d'aquesta llibreria enfocada a la representació de les dades d'enviaments transfronterers de residus. Per altra banda, un objectiu també és aprendre el trasfons teòric que comprèn l'anàlisi de xarxes per mitjà del desenvolupament pràctic d'aquestes amb el programari lliure R.

L'anàlisi de xarxes (*Network Analysis* en anglès) no té una definició concreta, però a Brandes i Erlebach (2005) es diu que des del punt de vista de la informàtica, es podria incloure en la “teoria de grafs aplicats”, ja que els aspectes estructurals i algorítmics dels grafs abstractes són els determinants prevalents en moltes aplicacions, independentment del tipus de xarxes que s'estigui modelant. Segons Kolaczyk i Gábor (2020), la imatge d'una xarxa –és a dir, essencialment, una cosa que s'assembla a una xarxa– s'utilitza per captar la noció d'elements d'un sistema i la seva interconnexió.

El tractament de dades i les anàlisis realitzades en aquest treball han estat elaborades mitjançant el programari lliure R, així com també ho ha estat la web app interactiva Shiny, que es troba allotjada en la següent pàgina web: <https://esther-amores.shinyapps.io/network-analysis/>.

Per tal de garantir la reproduïbilitat dels resultats obtinguts, es pot trobar tot el codi utilitzat als annexos situats al final del document i, juntament amb les dades, també es pot trobar al següent usuari de GitHub  [esther-amores](#).

2 | Marc del treball

En aquest extens capítol es realitza una introducció del tema en què aquest treball es centra: l'anàlisi de xarxes. Es comença fent una breu explicació de l'origen de les xarxes i de quines formes es poden analitzar, seguit de l'exposició dels conceptes clau que cal conèixer per entendre la teoria de grafs i, per acabar, s'assenyalen altres informacions d'interès.

2.1 Les xarxes

El motiu pel qual les xarxes han tingut un interès elevat pot explicar-se fàcilment amb l'expressió de “vivim en un món connectat”. Des de les xarxes socials com Facebook fins la *World Wide Web* i l'internet, estem rodejats per milers d'exemples en els que interactuem els uns amb els altres. També passa a nivell social, on existeixen diferents institucions creades per l'ésser humà (com ara governs), processos (com per exemple l'economia) i infraestructures (la xarxa global de companyies aèries n'és un clar exemple). I, per suposat, els humans no som els únics que formem part de complexos sistemes interconnectats. Observant el món natural que ens envolta, veiem una gran quantitat d'exemples d'aquest tipus de sistemes, des d'ecosistemes sencers, fins a xarxes biològiques d'aliments, passant per col·leccions de gens que interactuen o neurones que es comuniquen. Així doncs, la imatge d'una xarxa –és a dir, essencialment, una cosa que s'assembla a una xarxa– s'utilitza per captar la noció d'elements d'un sistema i la seva interconnexió, i el terme *xarxa* s'utilitza en una gran varietat d'àmbits. L'*Oxford English Dictionary* defineix, en la forma més general, *xarxa* com a “una col·lecció de coses interconnectades”. Per altra banda, el terme *xarxa* és sinònim del terme *graf* ja que, pels propòsits matemàtics, les xarxes es representen mitjançant grafs.

L'origen de l'anàlisi basada en xarxes en l'àmbit de ciències, concretament els fonaments matemàtics de la teoria de grafs, se situa pels voltants de 1735, quan Euler trobà una solució al problema del pont de Königsberg, en què demostrava que era impossible caminar pels set ponts d'aquesta ciutat de tal manera que recorrés cadascun sola vegada. Des d'aleshores, sobretot des de mitjans del 1800, aquestes arrels han crescut en diverses àrees (Kolaczyk i Gábor, 2020).

2.2 Tipus d'anàlisi de xarxes

Les dades de xarxes es recopilen diràriament en diverses àrees. Cadascuna d'aquestes té les seves pròpies qüestions i problemes sota estudi. No obstant, des d'una perspectiva estadística hi ha un fonament metodològic, compost de tasques i eines que son comuns a

alguns subconjunts no trivials d'àrees de recerca relacionades amb la ciència de xarxes. A més a més, és possible (de fet, és molt útil) categoritzar moltes de les diverses tasques que s'enfronten en l'anàlisi de les dades de xarxa entre diferents dominis segons una taxonomia estadística. Així doncs, en la línia d'aquesta taxonomia es passa dels mètodes descriptius als mètodes de modelatge i inferència, amb aquests últims separats, respectivament, en dues subàrees que corresponen al modelatge i a la inferència de les pròpies xarxes enfront dels processos a les xarxes (Kolaczyk i Gábor, 2020).

2.2.1 Visualització i caracterització de dades

Realitzar una anàlisi descriptiva de les dades, és a dir: la visualització i caracterització numèrica d'una xarxa, és una de les primeres passes en l'anàlisi de xarxes. De fet, a la pràctica les anàlisis descriptives constitueixen possiblement la majoria dels treballs de *Network Analysis* publicats. Els grafs representen les xarxes, i son molt visuals quan es tracten xarxes petites. En el cas de xarxes molt grans, com per exemple les xarxes socials, és impossible mostrar aquestes xarxes de manera que els usuaris i les amistats siguin evidents. En el cas de la biologia passaria el mateix per a xarxes de proteïnes i la seva afinitat per acoplar-se, per exemple, o també pels gens i les seves relacions reguladores (ib.).

La descripció de les dades mitjançant resums numèrics és un altre aspecte important de l'anàlisi descriptiva per a xarxes. S'utilitzen mesures de centralitat bàsiques, com ara la mitjana i la mediana; mesures de dispersió, com la desviació estàndard i el rang; la transitivitat (també anomenada coeficient d'agrupació *clustering coefficient*), que mesura la probabilitat que els nodes adjacents a un node estiguin connectats (Gábor, 2021); i l'assortativitat (o coeficient d'assortativitat *assortativity coefficient*), que és la preferència dels nodes d'una xarxa a unir-se amb d'altres que son similars en alguna característica, i es quantifica amb un tipus de correlació (Wikipedia, 2021a).

Cal destacar que la transitivitat de la xarxa es pot calcular numèricament mitjançant una enumeració de la proporció de triplets de vèrtexs que formen **triangles**, que son 3 nodes connectats per 2 arestes (Kolaczyk i Gábor, 2020).

2.2.2 Modelització i inferència de xarxes

Més enllà de preguntar-nos com és una xarxa observada i caracteritzar la seva estructura, pot interessar comprendre com ha sorgit. Es pot concebre la xarxa com el resultat d'alguns processos subjacents associats al complex sistema d'interès i preguntar-nos quins son els aspectes essencials d'aquests processos segons Kolaczyk i Gábor. A més a més, la manera en què la xarxa s'ha obtingut, és a dir: el procés de mesura i construcció corresponents, pot ser un aspecte molt rellevant a tenir en compte. Aquests fets proporcionen l'impuls per al modelatge de xarxes i per a cercar eines d'inferència estadística.

A grans trets, hi ha dues classes de models de xarxes: els **matemàtics** i els **estadístics**. Els **models matemàtics** son aquells que han estat especificats a través de regles probabilístiques per generar una xarxa, on sovint les normes son definides en un intent de capturar un mecanisme en particular o un principi (per exemple, “el ric és fa més ric”). En canvi, els **models estadístics** també son sovint models probabilístics, especificats amb la intenció que siguin adequats a les dades observades, cosa que

permet, per exemple, l'avaluació del poder explicatiu de certes variables en la formació d'enllaços a la xarxa, així com també que l'ajust de les dades s'efectuï i s'avaluï utilitzant els principis formals d'inferència estadística. Tot i que certament hi ha una certa superposició entre aquestes dues classes de models, les publicacions rellevants son, però, en gran mesura diferents.

L'exemple més senzill d'un model de xarxa matemàtica és aquell en què els seus arcs son assignats aleatòriament a parelles de nodes basats en el resultat d'una col·lecció de llançaments de monedes independents i idènticament distribuïts. És a dir, un llançament per a cada parell de nodes. Aquest model correspon a una variant de la fórmula d'un graf aleatori d'Erdős–Rényi, el qual ha estat extensivament estudiat des de 1960.

Des de la perspectiva estadística, en canvi, aquests models matemàtics son massa simples per a un bon ajust a les dades reals de la xarxa. No obstant, no només son útils per permetre obtenir una visió formal sobre com els mecanismes específics de formació d'enllaços poden afectar l'estructura de la xarxa, sinó que també s'utilitzen habitualment per definir classes de xaxes nul·les contra les quals avaluar la significació de les característiques estructurals trobades en una xarxa (Brandes i Erlebach, 2005).

2.3 Teoria de grafs

2.3.1 Fonaments

Prenem el terme xarxa com el concepte informal que descriu un objecte compost d'elements i interaccions o connexions entre aquests elements. Aleshores, un **graf** $G = (V, E)$ és una estructura matemàtica que consisteix en un conjunt V de **vèrtexs** (també anomenats **nodes**) i un conjunt E d'**arestes** (o **enllaços**) que uneixen (també es diu que connecten) parelles de vèrtexs. El conjunt de nodes i el conjunt d'enllaços es denoten com a $V(G)$ i $V(E)$, respectivament (ib.).

El número de vèrtexs $N_v = |V|$ i el número d'enllaços $N_e = |E|$ de vegades s'anomenen l'**ordre** i la **mida** del graf G , respectivament. Sovint, i sense pèrdua de la generalitat¹, s'etiqueten els vèrtexs simplement amb els enters $1, \dots, N_v$ i les arestes de manera anàloga: E_1, \dots, E_e (Kolaczyk i Gábor, 2020).

Els dos vèrtexs units per una aresta es diu que son els **vèrtexs finals** de l'aresta. Si dos vèrtexs estan units per una aresta, son **adjacents** i s'anomenen **veïns**.

Els grafs poden ser **dirigits** o **no dirigits**. En els grafs no dirigits, l'ordre dels vèrtexs finals és irrellevant i una aresta no dirigida que uneix els vèrtexs $u, v \in V$ es denota per u, v . En els grafs dirigits, cada aresta dirigida s'anomena **arc** que té un origen (**cua**) i una destinació (**cap**) i, per tant, una aresta amb origen $u \in V$ i una destinació $v \in V$ es representa per la parella ordenada (u, v) . Així doncs, l'enllaç u, v també es pot denotar com a uv i com a vu , mentre que el parell (u, v) pot anomenar-se uv . Amb menys freqüència també es poden trobar grafs que tenen enllaços tant dirigits com no dirigits: els **grafs mixtos** (Brandes i Erlebach, 2005).

També hi ha altres tipus de grafs, com els que es descriuen en els següents subapartats.

¹Un graf $*G*$ és únic fins a les noves etiquetes dels seus vèrtexs i nodes que deixen l'estructura inalterada. Dos gràfics equivalents en aquest sentit s'anomenen **isomorfs**.

Multi-grafs

Tant en els grafs dirigits com en els no dirigits, podem permetre que el conjunt d'arestes E contingui la mateixa aresta diverses vegades, és a dir, E pot ser un conjunt múltiple. Això és el que s'anomena **multi-graf**. Si es produeix una aresta diverses vegades a E , les còpies d'aquesta s'anomenen **arestes paral·leles** (els grafs amb aquest tipus d'arestes també s'anomenen multi-grafs). Un graf és **simple** si cadascuna de les seves arestes està continguda en E només una vegada, és a dir, si el graf no té arestes paral·leles. Una aresta que uneix un vèrtex a si mateixa, és a dir: una aresta els vèrtexs finals de la qual són idèntiques, s'anomena **bucle**. Un graf s'anomena **lliure de bucles** si no en té. Es suposarà que tots els grafs no tenen bucles tret que s'indiqui el contrari (Brandes i Erlebach, 2005).

Grafs ponderats

Sovint és útil associar valors numèrics (pesos) a les arestes o als vèrtexs d'un graf $G = (V, E)$, però per ara ens centrarem en els pesos de les arestes. Aquests grafs són anomenats **grafs ponderats**. Els pesos dels enllaços poden ser representats com una funció $\omega : E \rightarrow \mathbb{R}$ que assigna a cada enllaç $e \in E$ un pes $\omega(e)$. Depenent del context, els enllaços ponderats poden descriure diferents característiques com poden ser el cost (de viatjar o d'una distància, per exemple), la capacitat, la força d'interacció o la similitud. Normalment s'intenten indicar les característiques dels pesos de les arestes mitjançant l'elecció del nom per a la funció. En concret, una funció que assigna capacitats (superiors) a les arestes es denota sovint per u , especialment en el context dels problemes de flux de xarxa. En general, s'utilitzarà principalment ω per indicar enllaços ponderats que expressen capacitats o forces d'interacció. Per a la majoria de propòsits, un graf no ponderat $G = (V, E)$ equival a un graf ponderat amb pesos als enllaços unitaris $\omega(E) = 1$ per a tot $e \in E$ (ib.).

Graus

El **grau** d'un vèrtex v en un graf no dirigit $G = (V, E)$, denotat per d_v , és el nombre d'enllaços en E que tenen v com a vèrtex final. Si G és un multigraf, les arestes paral·leles es compten segons la seva multiplicitat en E . El conjunt d'enllaços que tenen v com a vèrtex final s'anomena $\Gamma(v)$. El conjunt de veïns de v es denota amb $N(v)$. En un graf dirigit $G = (V, E)$, els **graus externs** (*out-degree*) de $v \in V$, denotats com d_v^{out} , són el nombre d'arestes en E que tenen com a origen v . Els **graus interns** (*in-degree*) de $v \in V$, denotats com d_v^{in} , són el número d'arestes que tenen com a destinació v . Pels grafs ponderats, tots aquests conceptes es generalitzen sumant els pesos de les arestes enlloc de prendre el seu número.

El grau mitjà es denota amb $\bar{d}(G) = \frac{1}{|V|} \sum_{u \in V} d(u)$. Un graf no dirigit $G = (V, E)$ s'anomena **regular** si tots els seus nodes tenen el mateix grau, i **d-regular** si aquest grau és igual a d (ib.).

Subgrafs

Un graf $G' = (V', E')$ és un **subgraf** del graf $G = (V, E)$ si $V' \subseteq V$ and $E' \subseteq E$. El **subgraf induït pels vèrtexs** amb el conjunt de vèrtexs $V' \subseteq V$ denotat per $G[V']$, és el subgraf $G' = (V', E')$ de G , on E' és el conjunt de tots els enllaços $e \in E$ que

s'uneixen amb els vèrtexs en V' . El **subgraf induït pels enllaços** amb el conjunt d'enllaços $E' \subseteq E$, denotat per $G[E']$, és el subgraf $G' = (V', E')$ de G , on V' és el conjunt de tots els vèrtexs en V que són vèrtexs finals d'almenys un enllaç en E' . Si C és un subconjunt adequat de V , aleshores $G - C$ denoten el graf obtingut de G eliminant tots els vèrtexs en C i els seus enllaços incidents. Si F és un subconjunt d' E , $G - F$ denoten el graf obtingut de G eliminant tots els enllaços en F (Brandes i Erlebach, 2005).

2.3.2 Components connectats

Un graf no dirigit $G = (V, E)$ està **connectat** si es pot arribar a tots els vèrtexs des de tots els altres vèrtexs, és a dir, si hi ha un camí des de cada vèrtex fins a tots els altres vèrtexs. Els grafs no connectats s'anomenen **desconnectats**. Per a un graf no dirigit $G = (V, E)$, un **component connectat** de G és un subgraf induït $G' = (V', E')$ que està connectat i és màxim (és a dir, no hi ha cap subgraf desconnectat $G'' = (V'', E'')$ amb $V'' \supset V'$).

Un graf dirigit $G = (V, E)$ està **fortament connectat** si hi ha un camí directe des de tots els vèrtexs fins a tots els altres vèrtexs. Un **component fortament connectat** d'un graf dirigit G és un subgraf induït que està fortament connectat i és màxim. Un graf dirigit està **feblement connectat** si el seu graf no dirigit subjacent (és a dir, el resultat de treure el cap i la cua de G) està connectat (ib.).

Un graf **complet** és aquell on cada vèrtex està unit a tots els altres per un enllaç. Un **clique** és un subgraf complet.

Un graf connectat sense cicles s'anomena **arbre**. La unió disjunta d'aquests gràfics és un **bosc**. Els arbres tenen una importància fonamental en l'anàlisi de xarxes perquè serveixen com a estructura de dades clau en el disseny eficient de molts algorismes computacionals. Un graf dirigit que tingui un arbre cim a graf subjacent s'anomena **arbre dirigit**. Sovint, aquests arbres tenen associat un vèrtex especial anomenat **arrel**, que es distingeix per ser l'únic node des del qual hi ha un camí dirigit cap a tots els altres nodes del graf. Aquest graf s'anomena **arbre amb arrels**. Un vèrtex que precedeix un altre en un camí des de l'arrel s'anomena **ancestre**, mentre que un vèrtex que segueix un altre s'anomena **descendent**. Els ancestres immediats són els **pares** i els descendents immediats s'anomenen **fills**. Un node sense fills s'anomena **fulla**, i la distància des de l'arrel fins a la fulla més llunyana és la **profunditat de l'arbre**.

Una generalització important del concepte d'arbre és la d'un graf acíclic dirigit (**DAG**) que, tal i com diu el seu nom, és un graf dirigit sense cicles. Tanmateix, a diferència d'un arbre dirigit el seu graf subjacent no és un arbre, ja que substituir els arcs per nodes no dirigits deixa un graf simple que conté cicles (Kolaczyk i Gábor, 2020).

2.3.3 Distàncies i camins

Un **passeig** des de x_0 fins a x_k en un graf $G = (V, E)$ és una seqüència alternativa $v_0, e_1, v_1, e_2, v_2, \dots, v_{l-1}, e_l, v_l$ d'arestes i nodes, on $e_i = \{v_{i-1}, v_i\}$ en el cas indirecte i $e_i = (v_{i-1}, v_i)$ en el cas directe. La llargada del passeig l es defineix com el número d'enllaços del passeig. Un **sender** és un passeig sense enllaços repetits. Un **circuit** és un sender que comença i acaba pel mateix node. El passeig s'anomena **camí**, si $e_i \neq e_j$

per a $i \neq j$, i un camí és un **camí simple** si $x_i \neq x_j$ per a $i \neq j$. És a dir, un camí és un sender sense nodes repetits. Un camí amb $x_0 = x_k$ és un **cicle**. Un cicle és un **cicle simple** si $x_i \neq x_j$ per a $0 \leq i < j \leq k-1$, i si un graf no té cicles s'anomena **acíclic**.

Es diu que el vèrtex v en un graf G és **accessible** des d'un altre vèrtex u si hi ha un passeig d' u a v .

Per a un camí p en un graf $G = (V, E)$ amb pesos ω als enllaços, el pes del camí, denotat per $\omega(p)$, és definit com la suma dels pesos dels enllaços a p . Un camí des de u fins a v en G és el **camí més curt** (respecte a ω) si el seu pes és el més petit possible d'entre tots els camins que van des de u fins a v .

La **distància** entre vèrtexs en un graf es defineix com la longitud del camí (o camins) més curt (o curts) entre els vèrtexs (que s'igualen a infinit si no existeix aquest camí). Sovint s'anomena aquesta distància com a **distància geodèsica**, sent "geodèsic" un altre nom per als camins més curts. El valor de la distància més llarga d'un graf s'anomena **diàmetre** del graf (Kolaczyk i Gábor, 2020).

2.3.4 Teoria algebraica de grafs

La **matriu d'incidència** d'un graf dirigit $G = (V, E)$ amb $V = \{v_1, \dots, v_n\}$ i $E = \{e_1, \dots, e_m\}$ és una matriu B amb n files i m columnes que té entrades $b_{i,j}$ satisfent

$$\begin{cases} -1, & \text{si } v_i \text{ és l'origen de } e_j \\ 1, & \text{si } v_i \text{ és la destinació de } e_j \\ 0, & \text{altrament} \end{cases}$$

La **matriu d'adjacència** d'un graf dirigit simple $G = (V, E)$ amb $V = \{v_1, \dots, v_n\}$ és una matriu $A(G) = (a_{i,j})_{1 \leq i,j \leq n}$ de mida $n \times n$ amb

$$\begin{cases} 1, & \text{si } (v_i, v_j) \in E \\ 0, & \text{altrament} \end{cases}$$

Si s'expressa en paraules, A és diferent de 0 per a les entrades de (i, j) que corresponen als vèrtexs de G units per un enllaç, i 0 per aquelles que no ho son. Si G és un graf no dirigit, la seva matriu d'adjacència és simètrica i té $a_{i,j} = 1$ si v_i i v_j son adjacents. Pels grafs ponderats, les entrades diferents de 0 son $\omega(v_i, v_j)$ en lloc d'1 (Brandes i Erlebach, 2005).

2.4 Tractament de les dades

2.4.1 Informació sobre les dades

A continuació s'hi troba detallada la informació corresponent a la base de dades tractada en aquest treball que prové del portal Eurostat (*Waste shipments across borders* 2019).

La normativa i les directives de la UE sobre residus que s'enumeren a continuació requereixen l'enviament de dades dels Estats Membres a la Comissió Europea. Algunes d'aquestes directives de la UE han introduït objectius de recuperació i reciclatge sobre, per exemple, residus d'envasos, vehicles fora d'ús, bateries, residus d'aparells elèctrics

i electrònics, així com també sobre residus de construcció i demolició. Es necessiten dades sobre aquests fluxos de residus per controlar el compliment dels Estats Membres dels objectius establerts a les directives.

- Dades sobre quantitat, perillositat i enviaments de residus
 - Reglament d'estadístiques de residus (WStatR)
 - Reglament d'enviaments de residus (WShipR)
- Dades per al seguiment del compliment dels objectius
 - Directiva sobre bateries (BATT)
 - Directiva sobre vehicles al final de la vida útil (ELV)
 - Directiva sobre envasos i residus d'envasos (PACK)
 - Directiva sobre residus d'equips elèctrics i electrònics (WEEE)
 - Directiva marc de residus (WFD)
- Recopilació de dades conjunta amb l'OCDE ².
 - Residus municipals

El **reglament d'estadístiques de residus** té com a objectiu establir un marc per a la producció d'estadístiques comunes sobre la generació, recuperació i eliminació de residus. La informació sobre el tractament de residus es divideix pel tipus de tractament (recuperació, incineració amb recuperació d'energia, altres incineracions, eliminació a terra i tractament de terres) i per categories de residus. Tots els valors es mesuren en tones de residus. Els Estats Membres poden decidir lliurement els seus mètodes de recollida de dades. Les principals opcions són: enquestes, fonts administratives, estimacions estadístiques o una combinació de mètodes.

Els informes segons el reglament d'estadístiques de residus van ser revisats el 2010 pel Reglament de la Comissió (UE) núm. 849/2010). La millor estructura d'informes s'aplicà per primera vegada el 2012 (per a l'any de referència 2010), on s'hi afegixen alguns canvis com ara la introducció de noves categories de residus i la reorganització d'aquestes ja existents.

La quantitat de residus generats és un indicador important, per exemple, per controlar els patrons de consum o l'eficiència dels recursos de la producció industrial. Es pot utilitzar per mesurar un augment o disminució de residus al llarg del temps, i també la diferència de generació entre països. Els Estats Membres informen de les dades sobre la generació de residus en termes absoluts (per tones de residus generats). Eurostat calcula i presenta la generació de residus també en kg per habitant.

²L'Organització per a la Cooperació i el Desenvolupament Econòmic (OCDE) és una organització internacional que treballa per donar forma a polítiques que fomentin la prosperitat, la igualtat, les oportunitats i el benestar de tothom. Actualment, està formada per 38 països membres que s'estenen arreu del món, des d'Amèrica del Nord i del Sud fins a Europa i l'Àsia-Pacífic. Els països membres col·laboren amb els experts de l'OCDE, utilitzen les seves dades i anàlisis per informar de les decisions polítiques i juguen un paper clau en les revisions dels països membres, dissenyades per fomentar un millor rendiment. La Comissió Europea participa en el seu treball, però no té dret a vot. Espanya s'hi va adherir el 1961 (*Who we are 2020*)

2.4.2 La base de dades

La base de dades original tractada en aquest treball es titula *Transboundary Shipments of Waste*, està en format *.xlsx* i s'ha extret del portal Eurostat (*Waste shipments across borders 2019*), concretament del segon desplegable de la web. Aquest fitxer d'Excel conté dues fulles de càlcul, la descripció de les quals és troba tot seguit.

Eurostat table for website

Les vuit primeres files descriuen de manera breu i concisa de què tracta la base de dades que es troba a partir la novena fila. Expliquen que les dades estan reportades d'acord amb el reglament (EC) N^o 1013/2006 del Parlament Europeu i del Consell de la UE; que s'inclouen les dades "en cru" informades directament al Conveni de Basilea sobre el control dels moviments transfronterers de residus perillosos i la seva eliminació; i que la classificació en perillosos, no perillosos i altres categories de residus es va realitzar basant-se en els codis i les descripcions presentades pels Estats Membres de la UE. Més endavant s'explicaran aquests punts detalladament. Cal destacar també que inclou la data de la darrera actualització, que és el 30 de novembre de 2020. Per tant, aquestes dades representa que son bastant riguroses donat que la data d'actualització és bastant pròxima a l'actual i gràcies a les polítiques de qualitat de l'Eurostat.

Explanation of columns

Per altra banda, en aquesta fulla s'hi troba l'explicació detallada per a cadascuna de les variables del conjunt de dades que apareix a la primera fulla de càlcul. A continuació, es troba una taula resum de les característiques d'aquestes variables. Per a més informació, la descripció completa de les variables es troba a l'annex A.

Variable	Descripció
<i>Year</i>	És l'any de referència de les dades.
<i>Import/export</i>	Indica si les dades son importacions o exportacions.
<i>Country reporting</i>	País que reporta la informació.
<i>Population</i>	Població del país que reporta.
<i>Quantity in tonnes</i>	Quantitat d'enviament de residus en tones.
<i>Quantity in kg per capita</i>	Quantitat d'enviament de residus en kg per capita.
<i>To or from country</i>	Si l'enviament de residus que es descriu és una exportació (importació), el país d'aquesta variable és el país receptor (d'enviament).
<i>Disposal and recovery code</i>	Operació de tractament per a la qual s'envien els residus.
<i>General Basel Convention code -Y code</i>	Tipus de residu segons el Conveni de Basilea.
<i>European List of Waste code</i>	Tipus de residu segons la Llista Europea de Residus.
<i>Detailed Basel Convention code or OECD decision code</i>	Informació detallada sobre els residus enviats segons el Conveni de Basilea i la Decisió C(2001)107.
<i>Basis for classification to hazardous and non-hazardous by CRI</i>	Classificació dels residus com a perillosos o no perillosos juntament amb el motiu de la classificació.
<i>UN hazardous class list</i>	Característiques perilloses segons la classificació de les Nacions Unides.
<i>Country of transit stated by a code</i>	Codi del país o països pels quals han passat els residus transfronterers enviats durant l'enviament del país exportador al país importador.
<i>Notes</i>	Notes addicionals sobre l'enviament proporcionades pel país reportant.

Taula 2.1: Explicació de les variables de la base de dades *Transboundary Shipments of Waste*. A la primera columna, s'hi troba el nom de la variable original, és a dir, en anglès. A la segona columna, hi ha descrita la informació resumida per a cadascuna de les variables.

2.4.3 Preprocessament de les dades

Per tal de dur a terme un anàlisi de qualitat i coherent amb les dades adquirides és necessari realitzar un preprocessament consistent de les dades. Es pot fer el seguiment d'aquest procés en l'Annex B, on es troba el corresponent codi R comentat. Cal destacar que per tal de poder fer una bona importació de les dades, s'ha copiat la base de dades de la primera fulla de càlcul del fitxer d'Excel a un nou fitxer anomenat `waste.csv` objecte que pren el nom `waste` en R.

Així doncs, per garantir la continuïtat de les dades s'han realitzat petits canvis a la base de dades durant la seva manipulació. Un d'ells ha estat unificar els països Sèrbia i Montenegro en una sola categoria, donat que al fitxer de dades original hi ha un registre que indica que a l'any 2005 Sèrbia és importador de residus, però Sèrbia no va esdevenir independent fins el 5 de juny de 2006 (Wikipedia, 2021d) i, a més a més, a partir de l'any 2007 també hi ha diferents registres que fan referència a Sèrbia i Montenegro.

Per altra banda, l'estat de Macedònia del Nord s'anomena Macedònia fins el febrer de 2019 (Wikipedia, 2021c). No obstant, però, en el fitxer de dades original tots els registres son fins l'any 2018 i tot i això anomenen a l'estat de Macedònia com a Macedònia del Nord. Per tant, s'unifiquen tots els registres que continguin Macedònia del Nord per Macedònia.

Altrament, hi ha 395 registres dels 156452 existents a la base de dades que a la columna de *To.or.from.country* el valor que hi aplica és "World - not allocated". Per tant, a tots aquests casos se'ls hi assigna la categoria "Not specified".

A banda d'aquests canvis generats per la base de dades, s'han realitzat d'altres per unificar els noms de tots els països del món obtinguts mitjançant la funció `map_data` de la llibreria `ggplot2` (objecte anomenat `world`) amb els noms dels països que apareixen a la base de dades `waste`, independentment si aquests països estan contemplats en `waste` o no. Estan desglossats en l'annex B.

Per poder visualitzar les dades en un mapamundi, és necessari tenir les coordenades de latitud i longitud de cadascun dels països. Donat que a `world` apareixen les coordenades que calen per a dibuixar el mapa, es necessiten obtenir d'una altre font les coordenades "de dins dels països" per dibuixar la procedència de les importacions i/o exportacions³. És per això que s'han utilitzat les dades `world.cities` del paquet `maps`, per tal d'obtenir les coordenades de les capitals dels països en un nou objecte anomenat `capitals`. De la mateixa manera que amb els altres datasets (`world` i `waste`), s'han realitzat canvis per unificar els noms dels tres datasets (veure annex B).

Un cop realitzat tot el procediment d'estandarització dels noms dels països dels tres conjunts de dades, s'ha d'elaborar una única base de dades amb les coordenades de tots els que apareixen al dataset `waste`. Cal tenir en compte, però, que per la categoria de "Not specified" les coordenades assignades han estat 0 tant per la latitud com per la longitud per a que es puguin representar en l'app web. A continuació, es tradueixen tots al català, com també cal fer amb les categories de la resta de variables. No obstant,

³Malgrat que en aquest treball només s'analitzin les dades de les importacions, tal i com s'ha explicat a la secció Selecció de variables, en la web app Shiny es poden visualitzar tant les importacions com les exportacions.

cal tenir en compte que els noms de les variables de **waste** s'han mantingut en anglès per facilitar la comprensió i la seva crida, però sí que s'han homogeneïtzat els noms amb la funció `tibble` (és a dir, s'han reescrit els noms de tal manera que siguin únics i sintàctics, això significa que s'han tret els espais i els caràcters especials). Per últim, es genera un subconjunt de **waste** per a les importacions de residus i un altre per a les exportacions. Un detall que cal tenir en compte és que a les taules de l'annex C, el país República Democràtica del Congo s'ha abreujat com RD del Congo per millorar l'estètica d'aquestes.

2.4.4 Selecció de variables

Cal tenir en compte la descripció específica de la variable *Country reporting* (veure annex B). Hi ha diferències en la presentació d'informes entre importacions i exportacions, tal i com indica el mateix Eurostat a la seva base de dades. Això vol dir que els registres resportats per un país que importa residus, per exemple, no coincideixen en tots els aspectes amb els respectius països que exporten aquests residus.

Per altra banda, en termes generals, les importacions tenen 83072 registres mentre que les exportacions en tenen 73380. Si s'agrupen les dades per any i país, el conjunt de dades d'exportacions també té un número de files més elevat que el d'importacions ($2355 > 1026$). És per totes aquestes raons que l'anàlisi de xarxes es realitzarà sobre la font de dades més completa, en aquest cas es tracta de les importacions. De totes maneres, més endavant es pot veure que s'han considerat totes les dades tant d'importacions com exportacions per a visualitzar-les a l'aplicació web.

3 | Anàlisi de xarxes

Un graf per si sol, és a dir: una sola col·lecció de nodes i enllaços, no és suficient per a representar un sistema complex, donat que hi pot haver més informació donada en forma de variables que poden ser indexades pels nodes (per exemple, el número d'habitants per país que reporta la informació sobre l'importació de residus) o les arestes (per exemple, la quantitat de residus en tones). Alternativament, a un nivell més gran de granularitat, pot ser convenient associar vèrtexs o arestes amb grups per veure les relacions. La llibreria d'R `igraph` conté funcions per a una implementació senzilla i uns algorismes que permeten representar grans grafs de l'ordre de milions de vèrtexs i arestes (Kolaczyk i Gábor, 2020).

3.1 Introducció

Per tal de crear el graf dirigit $G = (V, E)$ a partir del dataset d'importacions de residus, s'utilitzarà la funció `graph_from_data_frame` de la llibreria `igraph` per representar convenientment tant els vèrtexs com els atributs dels enllaços, on V són tots els països (variables `From` i `To` de l'objecte `import`) filtrats per l'últim any de les dades: 2018, i E són els viatges dels residus des dels països de procedència fins als de destí amb atributs com la quantitat de residus en tones, la quantitat de residus per capita en kg, el codi d'eliminació i recuperació dels residus i el codi general de la convenció de Basilea. Cal tenir en compte que `example` és un objecte de classe `data frame` que conté tant els vèrtexs com els enllaços, i que l'argument `directed = TRUE` perquè es vol que el graf sigui dirigit.

Un cop l'objecte `g.import` és de classe `igraph`, es pot obtenir un resum amb la funció `summary`, on la primera línia de l'output explica característiques de la xarxa: la **D** indica que és un graf dirigit i la **W** diu que està ponderat. En quant als atributs, la (g/c) significa que és un atribut de caràcters a nivell del graf, la (v/c) indica que és un atribut de caràcters a nivell del vèrtex i la (e/n) vol dir que és un atribut numèric a nivell dels enllaços:

```
> summary(g.import)
IGRAPH 35ef514 DNW- 108 6060 --
+ attr: name (v/c), Pop_to (v/n), id (v/c), shadow (v/l),
      borderWidth (v/n), color.background (v/c),
| color.border (v/c), group (v/c), margin (v/n), weight (e/n),
      Quantity.in.kg.per.capita_to (e/n),
| hazardous (e/n), id (e/n), width (e/n), color (e/c), arrows (e/c),
      smooth (e/l), shadow (e/l)
```

A banda d'això, també es pot accedir als seus atributs mitjançant el símbol \$. A més a més, es pot trobar tant el número de vèrtexs $N_v = 108$ com el número d'enllaços $N_e = 6060$ (aquestes dades també apareixen en l'output del `summary`), així com també si es tracta d'un multigraf o d'un graf simple, si és dirigit o no, si està ponderat o no, i si està connectat o no.

Obtenim que G es tracta d'un multigraf. Abans ja s'havia configurat com a dirigit, així que l'output obtingut amb la segona funció és correcte. A més a més, és un graf ponderat perquè s'ha ponderat segons la variable que indica el pes dels residus en tones i, per últim, està connectat perquè s'arriba a cada vèrtex des de tots els altres.

Si s'aprofunditza en el tipus de connexió dels components, el graf està feblement connectat perquè el seu graf subjacent està connectat.

3.2 Visualització

Les tècniques per mostrar grafs són el focus del camp del dibuix de gràfics o de la visualització de gràfics. Aquestes tècniques solen intentar incorporar una combinació d'elements de matemàtiques, estètica humana i algorismes.

Suposem que tenim un conjunt de mesures de xarxa que s'han codificat en una representació gràfica de xarxa $G = (V, E)$, i ara volem resumir G d'una manera visual. Un dels reptes de la visualització de grafs és crear representacions geomètriques d'estructures combinatòries utilitzant símbols (per exemple, punts, cercles, quadrats, etc.) per als vèrtexs $v \in V$ i corbes suaus per als enllaços $e \in E$. Per als humans, el més convenient és, per descomptat, dibuixar-ho en espai bidimensional, a diferència de l'espai tridimensional o en alguna superfície més abstracta. Per tant, restringirem la nostra atenció a aquesta configuració de dues dimensions (Kolaczyk i Gábor, 2020).

3.2.1 igraph

Amb el paquet anteriorment utilitzat `igraph` es poden dissenyar diferents gràfics amb la comanda `plot`, que crida una varietat d'algoritmes que, quan s'apliquen a un objecte de tipus `igraph`, es poden produir un assortiment bastant ric de visualitzacions. Els paràmetres associats a la trama permeten tant l'especificació de l'algoritme que s'utilitza com la configuració d'estètiques i restriccions.

La disposició més simple és la **circular**, on els vèrtexs estan situats de manera equidistant al voltant de la circumferència d'un cercle, i les arestes es dibuixen a través seu.

D'altres dissenys més eficaços són aquells basats en l'explotació d'analogies entre l'estructura relacional dels grafs i les forces entre els elements dels sistemes físics, s'anomenen mètodes d'**incorporació de salts** (*spring-embedder*). El primer enfocament d'aquesta àrea proposa introduir forces d'atracció i repulsió mitjançant l'associació de vèrtexs amb boles i arestes amb salts. Si s'interromp un sistema literal de boles connectades per salts, s'estènen alguns dels orígens i se'n comprimeixen d'altres i, en deixar-les anar, tornaran al seu estat natural. El mètode de Fruchterman i Reingol és un exemple habitual d'aques tipus de disposició.

De vegades, els grafs tenen una estructura especial que val la pena remarcar, com per exemple quan tenen estructura d'**arbre**. Si es representa en forma circular, no se

sap si el graf és circular o no. Tanmateix, tant amb la disposició radial de l'arbre com amb la disposició en capes, l'estructura és evident (Kolaczyk i Gábor, 2020).

Totes les disposicions explicades en aquesta secció es poden trobar a la figura 3.1. S'han omès les etiquetes dels nodes per tal que els grafs siguin més entenedors.

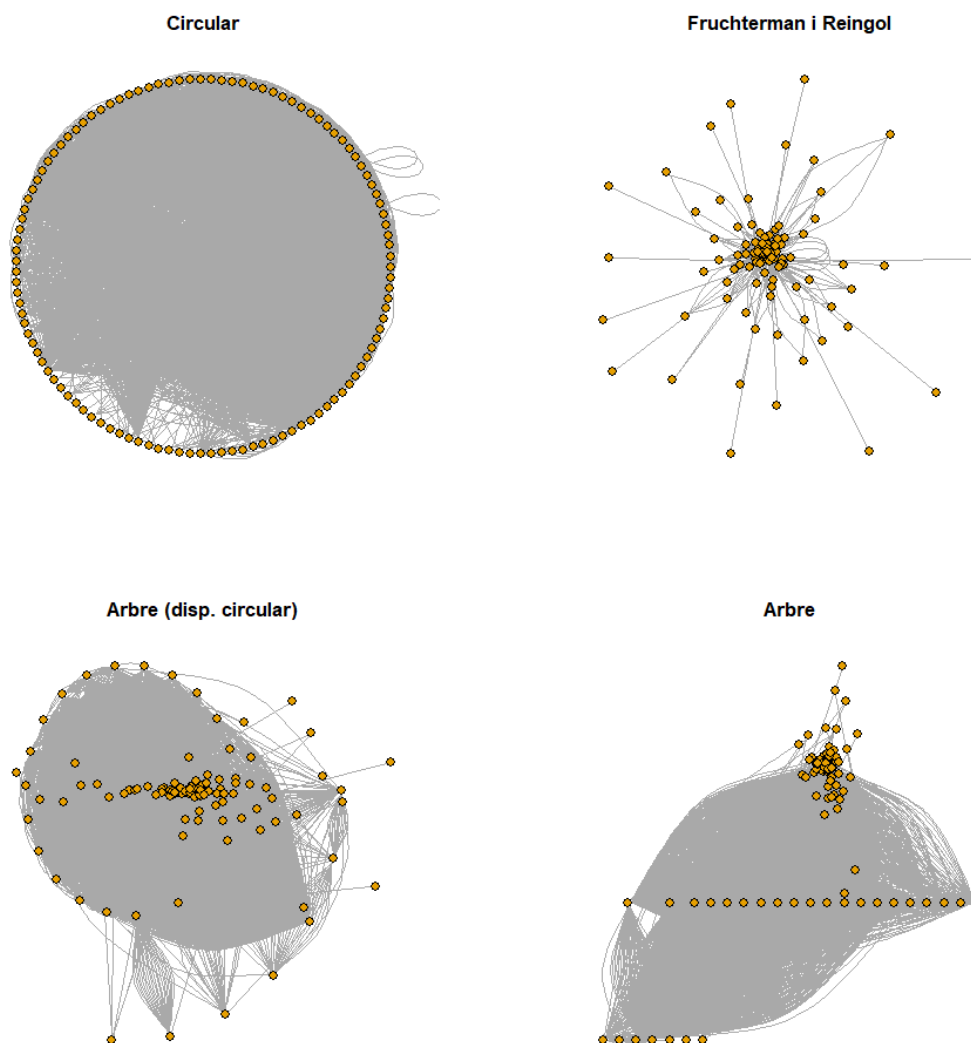


Figura 3.1: Grafs representats amb diferents disposicions de les importacions de tots el països l'any 2018

3.2.2 plotly i visNetwork

El paquet `plotly` és de codi obert i gratuït, i serveix per a crear gràfics interactius de gran qualitat pensats per a ser publicats en webs (*Plotly R Open Source Graphing Library* 2021).

Així doncs, amb el paquet `plotly` s'ha realitzat un mapa interactiu per a visualitzar de forma agradable i dinàmica les dinàmiques entre els diferents països en quant a la importació i exportació de residus transfronterers.

Hi ha moltes maneres de fer un mapa amb aquesta llibreria, però en termes generals es divideixen en dues grans categories: de forma integrada o personalitzada. En aquest cas, s'ha realitzat el mapa de forma integrada, doncs és més convenient si es necessita un mapa més “ràpid” i sense representacions sofisticades d'objectes geoespacials. Concretament, s'ha emprat la funció `plot_geo` que permet projectar el mapa des de diferents angles, inclosa la projecció ortogràfica que dona l'efecte del globus terraquí (Sievert, 2019). Aquí, s'hi poden representar marcadors amb la funció `add_markers` i segments (`add_segments`) que permeten visualitzar els trajectes desitjats.

La llibreria d'R `visNetwork` té com a funció visualitzar xarxes utilitzant la biblioteca JavaScript `vis.js`. És fàcil d'utilitzar, permet personalitzar les formes, estils, colors i mides dels grafs (igual que amb `igraph`), funciona sense problemes en qualsevol navegador modern i és compatible amb Shiny. No obstant, té límit en quant a la capacitat de representació de nodes i enllaços, que n'accepta fins a uns milers (*visNetwork, an R package for interactive network visualization* 2017).

3.2.3 Shiny

Per tal de facilitar la visualització dels grafs s'ha creat una aplicació amb Shiny, que és un paquet d'R que facilita la creació d'aplicacions web interactives directament des de R. Es poden allotjar aplicacions autònomes en una pàgina web o incrustar-les en documents R Markdown o crear taulers, així com també es poden ampliar les aplicacions amb temes CSS, htmlwidgets i accions de JavaScript (*Shiny* 2020).

L'aplicació es troba allotjada en la següent pàgina web: <https://esther-amores.shinyapps.io/network-analysis/>

Dins l'aplicació, es mostren dues opcions de visualització.

La primera es troba en el panell **Mapamundi**, on s'hi troba una representació de les dades plasmades sobre el globus terraquí realitzada amb `plotly`, així com també una taula dinàmica en base a les dades seleccionades. Seleccionant una categoria, un any i un (o varis) països de procedència, en el mapamundi apareixeran els diferents registres dibuixats amb nodes i enllaços. Si es passa el cursor per sobre els països, s'obtenen les característiques per a cadascun d'ells, segons si s'han seleccionat importacions o exportacions. Si s'han seleccionat importacions, aleshores apareixeran de color vermell els països importadors (és a dir, les destinacions), indicant la suma total de residus en tones; i de color blau els exportadors (és a dir, els orígens), mostrant la població del país segons l'any seleccionat, la quantitat de residus totals en tones i la quantitat de residus per capita que s'han importat. Si s'han seleccionat exportacions, aleshores serà l'anàlog de les importacions. Es pot capturar la imatge del mapamundi mitjançant els botons que apareixen a la part superior dreta del mateix, que venen per defecte amb el paquet `plotly`. La taula de la dreta accepta filtres mitjançant el nom del país, com també deixa ordenar els seus valors segons la variable desitjada. A més a més, els resultats (sense filtrar) poden ser descarregats en format `.csv`, `.xlsx` o bé `.tex`. La taula C.1 és un exemple de les importacions des de tots els països a l'any 2018.

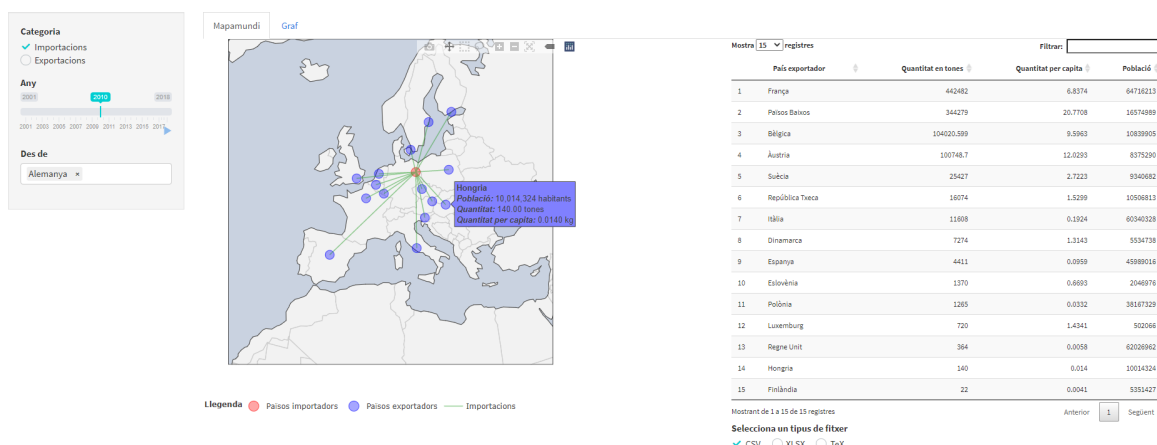


Figura 3.2: Captura de pantalla de l'aplicació Shiny on es mostra el mapamundi de les importacions des d'Hongria de l'any 2010.

La segona opció es troba en el panell **Graf**, i dóna una representació visual sobre els grafs, deixant triar quin tipus d'estructura es desitja (arbre, circular, estrella, Fruchterman i Reingol o estructura aleatòria). Si es fa *scroll* cap avall, es pot observar un botó de descàrrega que permet exportar la imatge del graf en format **.txt**.

Ambdós panells es controlen mitjançant el menú de l'esquerra, que serveix per a filtrar per categoria de les dades (importacions o exportacions dels residus), any i país de procedència (en cas que s'hagi triat veure les importacions) o destinació (en cas que s'hagi seleccionat les exportacions). Tanmateix, pel segon panell no s'hi representen les exportacions de residus pel motiu explicat anteriorment a la secció 2.4.4. Es poden diferenciar els residus perillosos (enllaços de color vermells) dels que no ho són (enllaços de color verdós). A més a més, com més gruixut sigui aquest enllaç, més quantitat de residus es transporten per a aquell moviment, tal i com es pot veure a la figura 3.3. A més a més, en aquesta figura

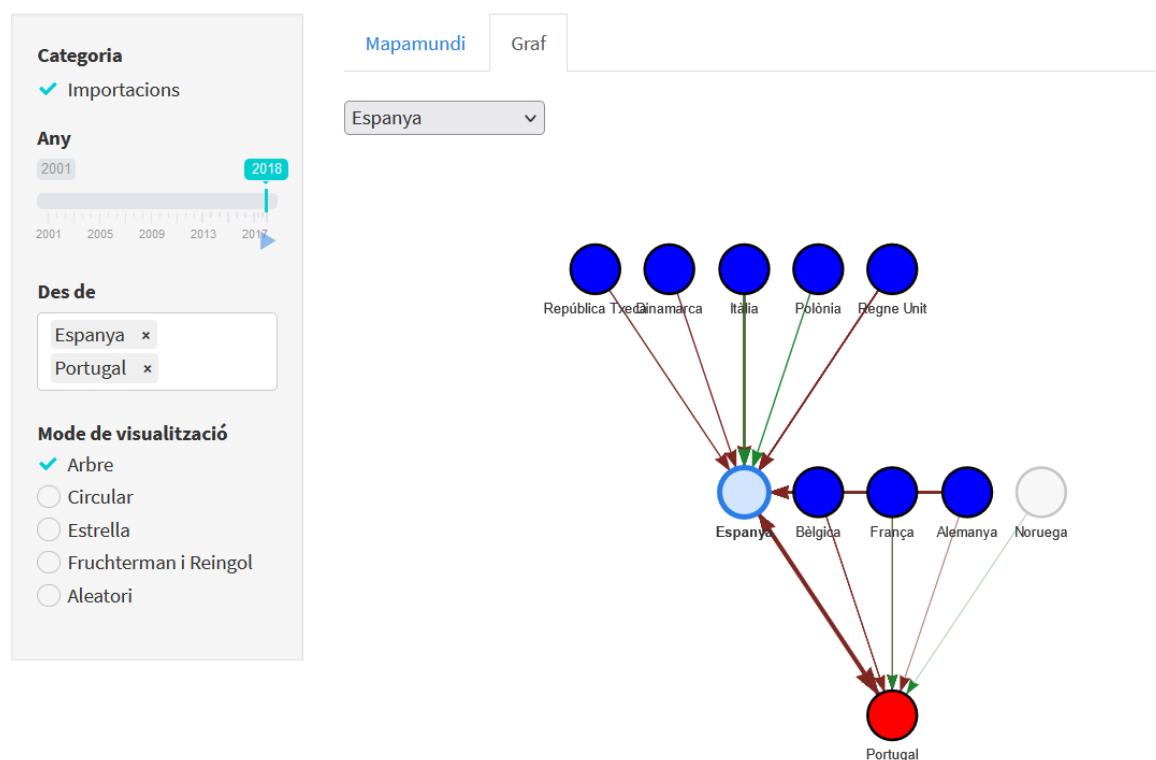


Figura 3.3: Captura de pantalla de l'aplicació Shiny on es mostra el graf de les importacions des d'Espanya i Portugal en disposició d'arbre de l'any 2018.

3.3 Anàlisi descriptiva

L'anàlisi de l'estructura dels grafs ha estat tradicionalment tractada com una tasca descriptiva en lloc d'una tasca inferencial, i les eines que s'utilitzen habitualment per a aquests objectius provenen en gran part d'àrees que no pertanyen a l'estadística convencional, segons Kolaczyk i Gábor. Per exemple, algunes d'aquestes eines són teoria de grafs i, per tant, tenen el seu origen en matemàtiques i informàtica. De la mateixa manera, el camp de l'anàlisi de xarxes socials també ha estat un altre component clau, contribuint amb eines que originalment anaven dirigides a captar aspectes bàsics de l'estructura i dinàmiques socials. Més recentment, el camp de la física també ha contribuït amb eines proposades sovint motivades per analogies de la mecànica estadística.

En aquesta secció s'aplicaran els conceptes anteriorment vistos a Teoria de grafs, així com també s'introduiran d'altres nous per donar lloc a l'anàlisi de les importacions de residus transfronterers des de tots els països que hi ha a la base de dades i filtrant per l'últim any que comprèn les dades, el 2018.

3.3.1 Característiques dels nodes i els enllaços

Donat que els elements fonamentals d'un graf són els seus enllaços i vèrtexs, hi ha nombroses característiques que es centren en aquests components i que concretament

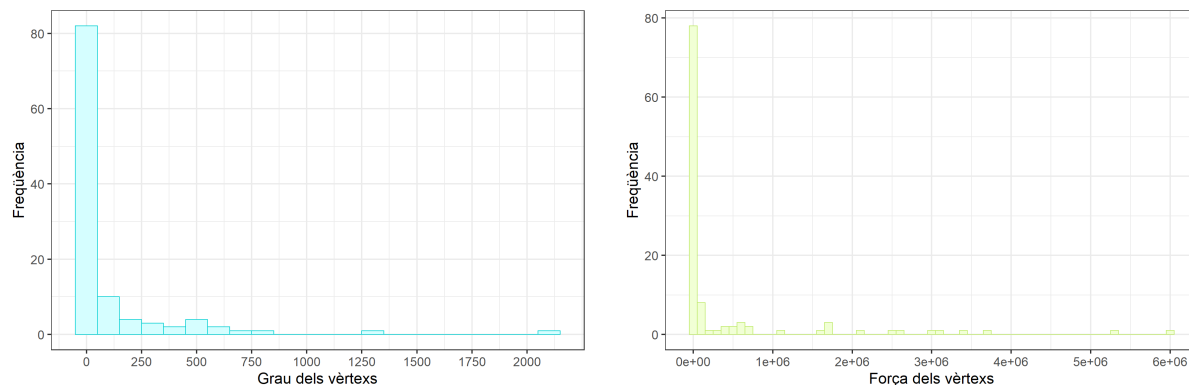


Figura 3.4: Distribució dels graus (a l'esquerra) i la força dels nodes (a la dreta) en importacions de tots el països l'any 2018

es divideixen en dos grans grups: descripcions basades en els graus dels nodes i mesures de centralitat dels nodes, que es centren en captar nocions generals de la “importància” dels vèrtexs.

Grau dels nodes

Tal i com hem vist anteriorment a Teoria de grafs, el grau d_v d'un vèrtex v , en un graf $G = (V, E)$, compta el número d'enllaços en E incidents sobre v . Donat un graf G , es denota f_d com la fracció de vèrtex $v \in V$ amb grau $d_v = d$. El conjunt $\{f_d\}_{d \geq 0}$ s'anomena **grau de distribució** de G , i consisteix en un canvi d'escala del conjunt de freqüències dels graus format a partir de la seqüència de graus original (Kolaczyk i Gábor, 2020).

A la figura 3.4(a) s'hi pot veure la distribució dels graus pel conjunt de dades tractat (és a dir, les importacions de residus) sense filtrar per cap país i de l'any 2018.

Es pot veure que bàsicament hi predomina un grup si es mesura per graus. Els nodes que estan més connectats corresponen a Alemanya ($d_v = 2441$), Bèlgica ($d_v = 1273$) i França ($d_v = 1215$), mentre que els que ho estan menys són països que no pertanyen a la Unió Europea, generalment, on només tenen 1 grau. Aquestes dades es poden trobar a la columna d_v de la taula C.2. Així doncs, hi ha una gran part de nodes amb un grau baix i també hi ha un nombre no trivial de vèrtexs amb graus de magnitud cada cop més elevada.

Donat que la xarxa tractada està ponderada segons la variable *Quantity in tonnes*, una generalització útil del grau és la força del vèrtex, que s'obté sumant els pesos de les arestes que es produeixen a un vèrtex determinat. La distribució de la força (o distribució de graus ponderats) es defineix per analogia amb la distribució de graus ordinària. Es pot trobar representada a la figura 3.4(b).

L'abast de la força del vèrtex no s'estén gaire més enllà del grau del vèrtex, així que la distinció prèviament feta entre els dos grups de vèrtexs (un amb més connectivitat que l'altre) no varia.

Més enllà de la distribució dels graus, resulta interessant trobar la manera en què els vèrtexs de diferents graus estan connectats entre si. És per això que és útil mirar la mitjana dels graus dels nodes veïns, tal i com es pot veure a la figura 3.5. Per

obtenir aquest gràfic, s'ha simplificat la xarxa de manera que s'han eliminat les arestes múltiples i els seus bucles.

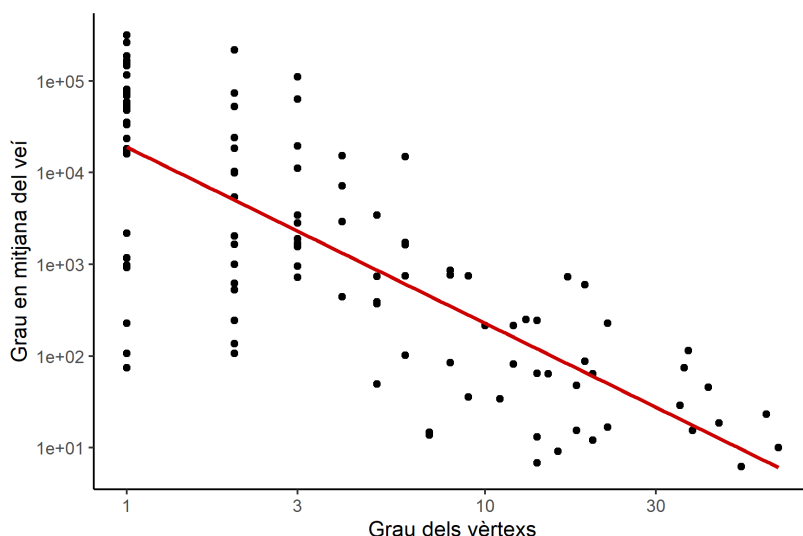


Figura 3.5: Grau mitjà del veí en relació al grau dels nodes (en escala log-log) en importacions des de tots els països l'any 2018

D'aquest gràfic se'n pot extreure que els vèrtexs de menor grau connecten amb els de major grau, mentre que els vèrtexs d'un ordre alt de graus tendeixen a associar-se amb els de menor grau. En color vermell s'hi troba representada la recta de regressió lineal, que indica que hi ha una relació negativa (o inversa), com s'ha comprovat.

Centralitat dels nodes

En una xarxa poden sorgir moltes preguntes relacionades amb la “importància” de cadascun dels nodes dins d'ella. Per exemple, pot servir per esbrinar quins són els països que realitzen més importacions de residus.

Les mesures de centralitat han estat dissenyades per quantificar aquestes nocions d’“importància” i, per tant, per facilitar-ne les respostes de les anteriors qüestions. En l'apartat anterior, ja s'ha trobat el que possiblement és la mesura més utilitzada de la centralitat dels vèrtexs: el grau. No obstant, hi ha una gran quantitat de maneres diferents per dur a terme aquests anàlisis, però en el llibre de Kolaczyk i Gábor es centren en les tres més comunes, que són la proximitat (*closeness*), l'interrelació (*betweenness*) i la centralitat del vector propi (*eigenvector centrality*).

En primer lloc, s'avalua la **centralitat de proximitat** (*closeness centrality*), que intenta captar si un vèrtex és “central” i si està “a prop” de molts altres nodes. L'enfocament aproximat, introduït per Sabidussi, consisteix en deixar que la centralitat varïi inversament amb una mesura de la distància total d'un vèrtex respecte tots els altres,

$$c_{Cl}(v) = \frac{1}{\sum_{u \in V} \text{dist}(v, u)}, \quad (3.1)$$

on $\text{dist}(v, u)$ és la distància geodèsica entre els vèrtexs $u, v \in V$. Sovint, per a la comparació entre grafs i amb altres mesures de centralitat, la centralitat de proximitat es normalitza a l'interval $[0,1]$, mitjançant la multiplicació per un factor $N_v - 1$.

Per altra banda, la **centralitat d'interrelació** (*betweenness centrality*) té com a objectiu resumir fins a quin punt es troba un vèrtex entre altres parelles de vèrtexs. Aquesta importància es refereix a on es troba un vèrtex respecte als camins (o rutes) del graf, així doncs els vèrtexs que se situen en molts camins són més crítics per al procés d'enviaments transfronterers de residus. La mesura de centralitat d'interrelació més comuna, introduïda per Freeman, es defineix com

$$c_B(v) = \sum_{s \neq t \neq v \in V} \frac{\sigma(s, t|v)}{\sigma(s, t)} \quad (3.2)$$

on $\sigma(s, t|v)$ és el número total de camins més curts entre s i t que passen per v , i $\sigma(s, t)$ és el número total de camins més curts entre s i t (independentment si passen per v o no). En cas que els camins més curts siguin únics, aleshores $c_B(v)$ només compara el número de camins més curts que passen per v . A més a més, es pot restringir a la unitat mitjançant la divisió per un factor de $(N_v - 1)(N_v - 2)/2$.

Per últim, les mesures de **centralitat del vector propi** (*eigenvector centrality*) es fixen més en el “rang” o “prestigi” perquè busquen captar la idea que com més centrals siguin els veïns d'un node, més central és el node mateix. són intrínscament implícites en la seva definició i normalment es poden expressar en termes de solucions de vectors propis de sistemes d'equacions lineals definits adequadament. Hi ha moltes mesures de centralitat del vector propi, com la de Bonanich, que la defineix com

$$c_{Ei}(v) = \alpha \sum_{u, v \in E} c_{Ei}(u) \quad (3.3)$$

El vector $c_{Ei} = (c_{Ei}(1), \dots, c_{Ei}(N_v))^T$ és la solució al problema del vector propi $\mathbf{A}c_{Ei} = \alpha^{-1}c_{Ei}$, on \mathbf{A} és la matriu d'adjacència per la xarxa G . Bonanich diu que una elecció òptima per a α^{-1} és el valor més gran de \mathbf{A} i, per tant, c_{Ei} és el corresponent vector propi. Quan G és un graf no dirigit i connectat, el valor propi més gran de \mathbf{A} serà simple i el seu vector propi tindrà entrades que seran diferents de 0 i tindran el mateix signe. Els valors absoluts de les entrades es situaran automàticament entre 0 i 1 per l'ortonormalitat dels vectors propis (Kolaczyk i Gábor, 2020).

En la figura 3.6 es poden veure les centralitats dels nodes per a les importacions des de tots els països de l'any 2018 obtingudes amb el paquet `sna`. En color groc s'hi troben dibuixats els nodes que tenen un grau més elevat, que corresponen als països Alemanya, França i Bèlgica, amb 70, 68 i 61 graus, respectivament. Cal tenir en compte que aquestes representacions són més aviat indicades per a grafs amb un número mitjà-baix d'arestes i nodes, per aquest motiu no s'aprecien gaire les mesures. No obstant, però, es pot veure que els nodes en groc estan molt centrats en el graf dels graus i en el de l'interrelació, mentre que en el del vector propi s'allunyen una mica més del centre. Per veure els resultats desglossats per a cadascuna de les mesures de centralitat, veure la taula C.4.

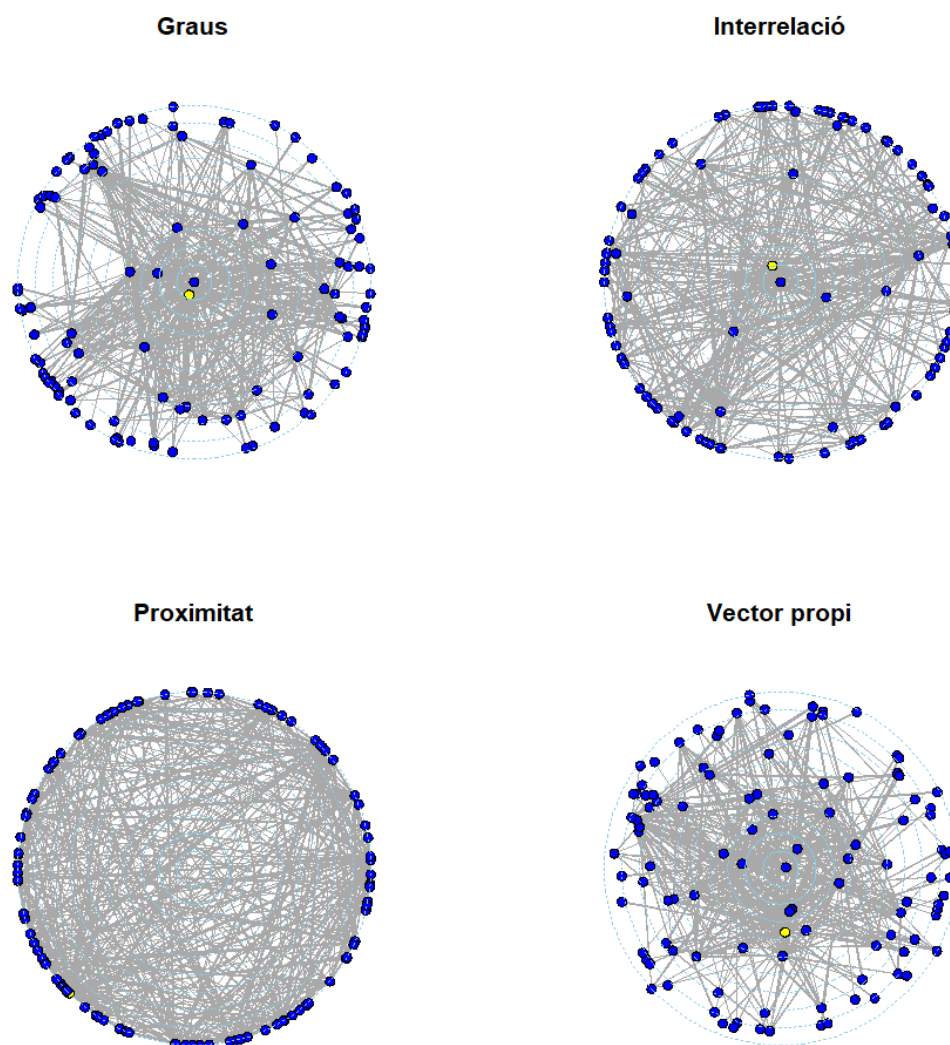


Figura 3.6: Gracs que mostren la centralitat dels nodes de la xarxa d'importacions de tots el països l'any 2018. Els països amb un major grau, Alemanya, França i Bèlgica, es mostren en color groc, mentre que la resta de nodes estan pintats de blau.

Caracterització dels enllaços

Fins ara només s'han mirat les característiques dels nodes, però ara també es mirarà la **centralitat d'interrelació** dels enllaços, assignant a cada aresta un valor que reflecteixi el número de camins més curts que la travessen a si mateixa. Així doncs, s'obté que per a tots els països importadors l'any 2018, el país que té un paper més important en quant al fluxe de la informació entre el Regne Unit i Espanya és Alemanya.

3.3.2 Cohesió de la xarxa

Hi ha moltes preguntes en l'anàlisi de xarxes que es resumeixen en qüestions relacionades amb la cohesió de la xarxa, la mesura en què els subconjunts de vèrtexs estan

cohesionats (o “units”) en relació als enllaços de la xarxa. Com a conseqüència, existeixen diverses maneres de definir la cohesió de la xarxa en funció del context de la pregunta que es faci. En aquest apartat s’analitzen una sèrie de maneres de definir i resumir la cohesió en un graf (Kolaczyk i Gábor, 2020).

Subgrafs i censures

Un enfocament per definir la cohesió de la xarxa és mitjançant l’especificació d’un determinat subgraf d’interès, concretament d’un clique, ja que son subgrafs complets i, per tant, son subconjunts de vèrtexs que estan totalment cohesionats en el sentit que tots els vèrtexs del subconjunt estan connectats per arestes. A la taula 3.1 s’hi pot trobar l’estructura del graf censurant per la freqüència de la mida dels cliques:

Mida del clique	1	2	3	4	5	6	7	8	9	10	11
Freqüència absoluta	108	386	941	1763	2344	2213	1474	673	197	32	2
Freqüència absoluta del màxim	–	41	19	13	12	9	15	9	1	11	2

Taula 3.1: Freqüència absoluta censurant per cliques i pel màxim d’aquests per la xarxa d’importacions des de tots els països de l’any 2018.

Hi ha 108 nodes (cliques de mida 1) i 386 enllaços (cliques de mida 2), seguits de 941 triangles (cliques de mida 3). Les cliques més grans son de mida 11, de les quals només n’hi ha 2 i tenen els següents països en comú: Alemanya, Itàlia, França, Regne Unit, Suècia, Països Baixos, Polònia, Bèlgica i Dinamarca. Aquesta anàlisi és una mica redundant perquè les cliques de mida més gran també inclouen les de mida més petita. Un **clique màxim** és aquell que no és subconjunt d’un clique més gran. Els 2 cliques de mida 11 son, per tant, màxims perquè no pertanyen a cap clique d’ordre superior.

Per altra banda, un ***k*-core** d’un graf G és un subgraf de G per al qual tots els graus del node son almenys k , de manera que cap altre subgraf que compleixi la mateixa condició el contingui, és a dir, és màxim en aquesta propietat. Aquesta noció de nuclis (*cores*) proporciona una manera de descomposar una xarxa en capes, que poden ser combinades d’una manera eficaç com pot ser la disposició radial que es presenta a la figura 3.7.

En ella, s’hi pot veure que els nodes amb menys nuclis, els colors vermells, taronges i grocs, es troben repartits als extrems del graf, i quan els nodes s’acosten més al centre del graf, els colors d’aquests son verdosos i blaus. És a dir, els vèrtexs amb $1, \dots, 46$ nuclis tenen una distància més curta entre ells que no pas els vèrtexs amb $47, \dots, 310$, que presenten una distància major. Els colors més blavosos son els nodes de nuclis $311, \dots, 402$, i son poc freqüents a la xarxa. Els vèrtexs amb un k -core major, amb $k = 597$, es troben situats al bell mig del graf i, segons la taula C.5 aquests nodes son Alemanya, França, Països Baixos i Bèlgica .

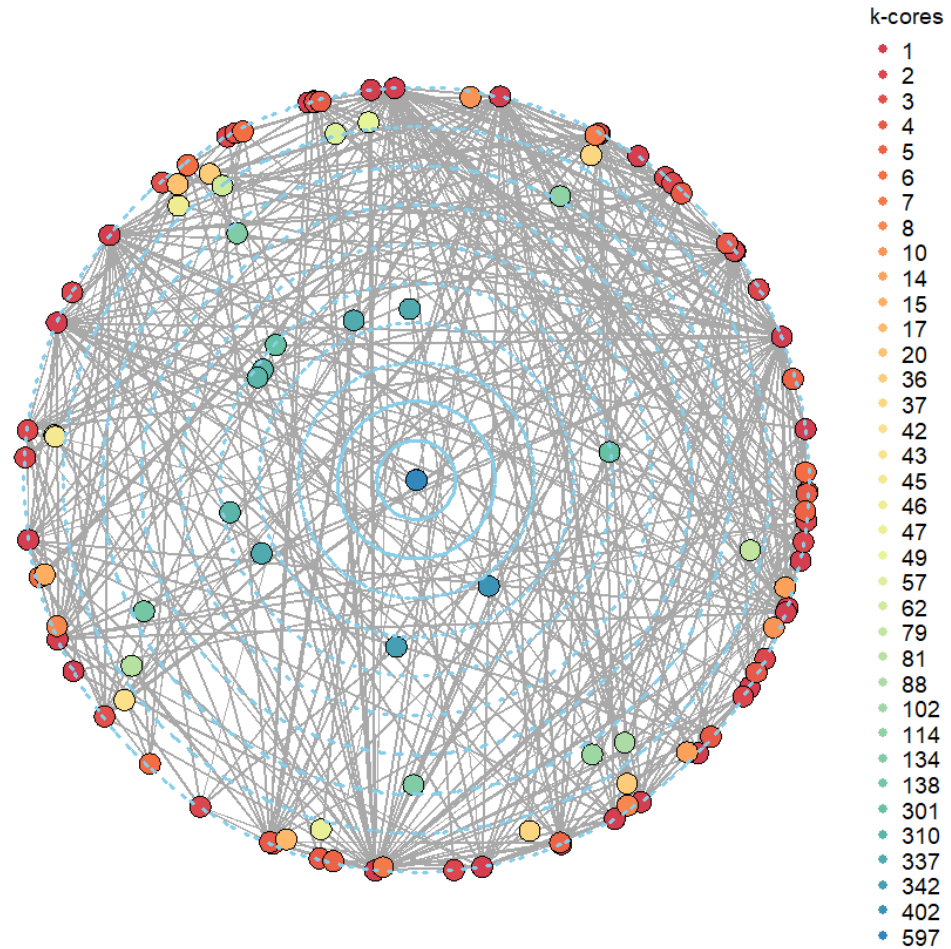


Figura 3.7: Representació visual de la descomposició *k-score* per la xarxa d'importacions des de tots els països de l'any 2018.

A part de tot això, analitzar les díades i les tríades pot ser fonamental. Les **díades** són parells de vèrtexs i, en grafs dirigits, poden adoptar tres estats: **nul** (sense arestes dirigides), **asimètric** (una aresta dirigida) o **mutu** (dues arestes dirigides). De la mateixa manera, les **tríades** són triplets de vèrtexs i poden adoptar setze estats, que van des del subgraf nul fins al subgraf en què les tres díades formades pels vèrtexs de la tríada tenen arestes mútues dirigides. Un cens dels possibles estats d'aquestes dues classes de subgrafs, és a dir, comptant quantes vegades s'observa cada estat en un graf G , pot donar una idea de la naturalesa de la connectivitat del graf (Kolaczyk i Gábor, 2020).

En el cas del subconjunt de dades analitzat, 5392 díades són nul·les i, de les que no són nul·les, 292 són asimètriques i 94 són mútues. Per tant, això indica unilateralitat en la manera en què els blocs es fan referència entre si.

Densitat i freqüència relativa

Les caracteritzacions de cohesió de xarxa fins ara descrites prenen com a hipòtesi que existeix una subestructura en la xarxa, i després miren si es produeix en un graf G i, si és així, amb quina freqüència. El concepte de freqüència relativa es pot aplicar de diverses maneres útils. La **densitat** d'un graf és la freqüència de les arestes obtingudes (les que estan en el graf) en relació amb les arestes potencials. Per exemple, en un graf no dirigit i simple (és a dir, sense bucles i sense enllaços múltiples), la densitat del subgraf $G' = (V', E')$ és

$$den(G') = \frac{|E'|}{|V'|(|V'| - 1)/2} \quad (3.4)$$

El valor de $den(G')$ recau entre 0 i 1 i proporciona una mesura de la proximitat de G' a ser un clique. En cas que G sigui un graf dirigit, aleshores el denominador es canvia per $|V'|(|V'| - 1)$.

El concepte de densitat és molt interessant gràcies a la llibertat que hi ha en l'elecció del subgraf G' que defineix (3.4). Per exemple, prenent $G' = G$ s'obté la densitat del graf global. Per contra, prenent $G' = G[V']$ com a conjunt de veïns d'un vèrtex $v \in V$, i els enllaços entre ells, es produeix una mesura de densitat al veí immediat de v (Kolaczyk i Gábor, 2020)¹. A part d'ajudar a entendre la connexió d'una xarxa en comparació amb la connexió que podria tenir, la densitat és útil per a comparar dues xarxes amb el mateix nombre de nodes i el mateix tipus de relacions, això indicarà en què es diferencien ambdues (Lizardo i Jilbert, 2020).

Per tant, el càlcul de la densitat per la xarxa dirigida d'importacions des de tots els països de l'any 2018 (és a dir, es pren $G' = G$) és:

$$den(G) = \frac{|E|}{|V|(|V| - 1)} = \frac{480}{108(108 - 1)} = 0.04153686$$

Aquest càlcul es pot realitzar “a mà” mirant la mida dels nodes i els enllaços, o bé amb la funció `edge_density` del paquet `igraph`. El resultat obtingut és un valor molt baix, pel que s'entén que la connexió de la xarxa global és dèbil: hi ha nodes que estaran molt connectats entre d'ells, sobretot entre els països Europeus com s'ha pogut observar en la figura 3.7.

La **freqüència relativa** també s'utilitza per definir nocions d'agrupació (*clustering*) en un graf. L'ús estàndard del coeficient d'agrupació fa referència a la quantitat

$$cl_T(G) = \frac{3\tau_\Delta(G)}{\tau_3(G)} \quad (3.5)$$

on $\tau_\Delta(G)$ és el número de triangles en el graf G , i $\tau_3(G)$ és el número de triplets connectats (és a dir, un subgraf de 3 nodes connectats per 2 arestes). El valor $cl_T(G)$ també es pot anomenar **transitivitat** del graf, com bé s'ha explicat al capítol 2.3. Així doncs, és una mesura de l'agrupació global que resumeix la freqüència relativa amb la què els triplets connectats es tanquen en triangles (Kolaczyk i Gábor, 2020). En la xarxa d'importacions des de tots els països de l'any 2018, obtenim amb la funció

¹La notació dels subgrafs que s'indica a Kolaczyk i Gábor s'ha substituït per la de Brandes i Erlebach.

transitivity que $cl_T(G) = 0.3275322$, això significa que gairebé un terç dels triplets connectats es tanquen.

Per altra banda, es pot obtenir el número de triplets connectats si es calcula el número de triangles existents en el graf amb la funció **triangles** i s'aïlla de l'equació 3.5:

$$cl_T(G) = \frac{3\tau_\Delta(G)}{\tau_3(G)} \Leftrightarrow 0.3275322 = \frac{3 \cdot 2823}{\tau_3(G)} \Leftrightarrow \tau_3(G) = 258527$$

Un altre concepte és el de la **reciprocitat**: fins a quin punt hi ha reciprocitat en una xarxa dirigida, és a dir, la proporció de connexions mútues. Hi ha dos enfocaments per a captar aquesta noció que es distingeixen segons si la unitat d'interès en el càlcul de les freqüències relatives són les **díades** o els **enllaços dirigits**.

En cas que s'utilitzin les **díades**, la reciprocitat es defineix com el nombre de díades amb arestes dirigides mútuament, dividides pel nombre de díades amb un enllaç únic i no recíproc. En aquest cas el valor és 0.3916667.

En cas que s'utilitzin els **enllaços dirigits**, la reciprocitat es defineix com el número total d'arestes mútues dividit pel nombre total d'arestes. En aquest cas el valor és 0.2435233.

Mirant els resultats obtinguts anteriorment de les díades, els baixos valors obtinguts de reciprocitat no són sorprenents donat que només n'hi havia 94 que fóssin mútues.

3.3.3 Connectivitat, talls i fluxos

Una qüestió bàsica d'interès és si un determinat graf es separa en subgrafs diferents. Si no és així, es pot intentar quantificar el grau de proximitat de fer-ho. Això és el que es veurà en aquest apartat.

Tal i com s'ha explicat a la introducció dels conceptes d'anàlisi de xarxes al capítol 2.3, un graf G està connectat si es pot arribar a tots els vèrtexs des de tots els altres vèrtexs, és a dir, si hi ha un camí des de cada vèrtex fins a tots els altres vèrtexs. Un component connectat de G és un subgraf induït $G' = (V', E')$ que està connectat i és màxim.

Sovint es dona el cas que un dels components connectats en un graf G domina els altres en magnitud, ja que conté la majoria dels vèrtexs en G , que és el que s'ha estat veient fins ara amb els països d'Alemanya, Bèlgica i França. Aquest component s'anomena **component gegant** (Kolaczyk i Gábor, 2020).

La xarxa d'importacions de residus des de tots els països l'any 2018 està connectada, com ja s'ha vist anteriorment.

Una característica observada en el component gegant de moltes xarxes és la propietat del **món petit**, que fa referència a la situació en què la distància del camí més curta entre parells de vèrtexs és bastant petita però l'agrupació és relativament alta.

En el graf analitzat, la longitud mitjana del camí del component gegant (que és l'únic component de la xarxa perquè és dirigida) és 2.093804, que és un valor molt baix. La mida del camí més llarg és 4, si no es contempla la xarxa ponderada amb els pesos de la variable de quantitat de residus en tones.

Per tant, la distància del camí més curt d'aquesta xarxa s'assembla més al logaritme de l'ordre del graf $\log(N_v) = 4.682131$ que a $N_v = 108$, així que es considera una distància petita. Alhora, l'agrupació d'aquesta xarxa és relativament gran en comparació, ja que la transitivitat és igual a 0.3275322. Això significa que gairebé un terç dels triplets connectats es tanquen en triangle, com ja s'ha vist abans.

3.3.4 Segmentació de grafs

La segmentació de grafs (*partitioning*, en anglès) consisteix en la partició d'un conjunt d'elements en subconjunts "naturals". Una partició $\mathcal{C} = \{C_1, \dots, C_k\}$ d'un conjunt finit S és una descomposició de S en K subconjunts disjunts no-buits tals que $\bigcup_{k=1}^K C_k = S$.

En l'anàlisi de xarxes, aquesta és una eina útil per trobar, de manera no supervisada, subconjunts de vèrtexs que demostrin una **cohesió** respecte els patrons relacionals subjacents. Un subconjunt **cohesionat** de vèrtexs es refereix al subconjunt de nodes que estan ben connectats entre ells i, alhora, estan relativament ben separats dels vèrtexs restants.

Aquest problema de la **segmentació de grafs** es coneix habitualment com a detecció de comunitats. Existeixen dues classes de mètodes per a treballar-la: els basats en adaptacions de l'agrupació jeràrquica i els basats en el particionament espectral. En aquest treball, només es realitzarà el primer enfocament (Kolaczyk i Gábor, 2020).

3.3.5 Agrupació jeràrquica

Molts mètodes per a la segmentació de gràfics són essencialment variacions del concepte més general d'agrupació jeràrquica (*hierarchical clustering*, en anglès) utilitzat en l'anàlisi de dades. Hi ha nombroses tècniques que s'han proposat pel problema de la segmentació i difereixen principalment en dos aspectes: en com s'avalua la qualitat de les agrupacions proposades i els algoritmes mitjançant els quals intenten optimitzar aquesta qualitat.

Els mètodes jeràrquics poden classificar-se de dues maneres, poden ser aglomeratius o divisius.

Els **aglomeratius** es basen en l'engrossiment successiu de particions mitjançant el procés de fusió o divisió, i s'executa la combinació menys costosa de dos elements de partició ja existents,

Els **divisius** es basen en el refinament successiu de particions mitjançant el procés de divisió, i s'executa la divisió menys costosa d'un sol element de partició.

En cada etapa, la partició candidata actual es modifica de manera que minimitza una funció de cost específica.

Una funció de cost força popular és la **modularitat**. Sigui $\mathcal{C} = \{C_1, \dots, C_k\}$ una partició candidata determinada i denotem $f_{ij} = f_{ij}(\mathcal{C})$ la fracció d'enllaços de la xarxa original que connecten nodes a C_i amb nodes a C_j . Aleshores, la modularitat de \mathcal{C} és el valor

$$mod(\mathcal{C}) = \sum_{k=1}^K [f_{kk}(\mathcal{C}) - f_{kk}^*]^2, \quad (3.6)$$

on f_{kk}^* és el valor esperat de f_{kk} sota algun model d'assignació d'enllaç aleatori.

Aquesta elecció correspon a un model en què es construeix un graf per tenir la mateixa distribució de graus que G , però amb arestes situades aleatòriament, sense tenir en compte els elements de partició subjacents dictats per \mathcal{C} . Per tant, es prenen grans valors de la modularitat per suggerir que \mathcal{C} captura una estructura de “grup” no trivial, més enllà del que s’espera que es produeixi sota l’assignació aleatòria d’enllaços.

En principi, l’optimització de la modularitat de 3.6 requereix cercar totes les particions possibles \mathcal{C} , però aquest és un procés molt car en xarxes de mida gran. Així doncs, s’optimitza mitjançant l’algoritme *fast and greedy* per obtenir un objecte de que representi les comunitats de classes (Kolaczyk i Gábor, 2020).

Un detall que cal tenir en compte és que la funció amb què s’executa aquest algoritme (`cluster_fast_greedy`) només accepta grafs no dirigits i simplificats.

Segons el que s’ha esbrinat d’aquesta xarxa anteriorment, i examinant la pertinença a la comunitat, s’obté la següent taula resum (veure taula 3.2) que s’ha realitzat a partir dels clústers obtinguts amb l’algoritme de *fast and greedy*, dels graus i dels nuclis dels nodes, és a dir, mostra un resum de la suma de graus i nuclis per a cadascun dels clústers. La taula sencera sense agrupar per clústers es troba a l’annex C, concretament a la taula C.6.

Clúster	Mida	$\sum d_v$	$\sum k$
1	27	5301	2449
2	38	3471	2372
3	24	2534	1895
4	19	814	653

Taula 3.2: Resum per a cada clúster on es presenta: la seva mida, la suma grau dels nodes i la suma nucli dels nodes per a les regions de la xarxa d’importacions des de tots els països de l’any 2018.

Es pot observar que el primer i el segon clúster son els que contenen els països que tenen un major número de graus i nuclis als nodes, a diferència de l’últim d’ells que té els de menor grau i nuclis. Per tant, és raonable pensar que els dos clústers més grans (1 i 2) contindran els països que tenen un major número de graus en els nodes i de nuclis, així com també una centralitat del vector propi major. La representació visual dels clústers es troba a la figura 3.8.

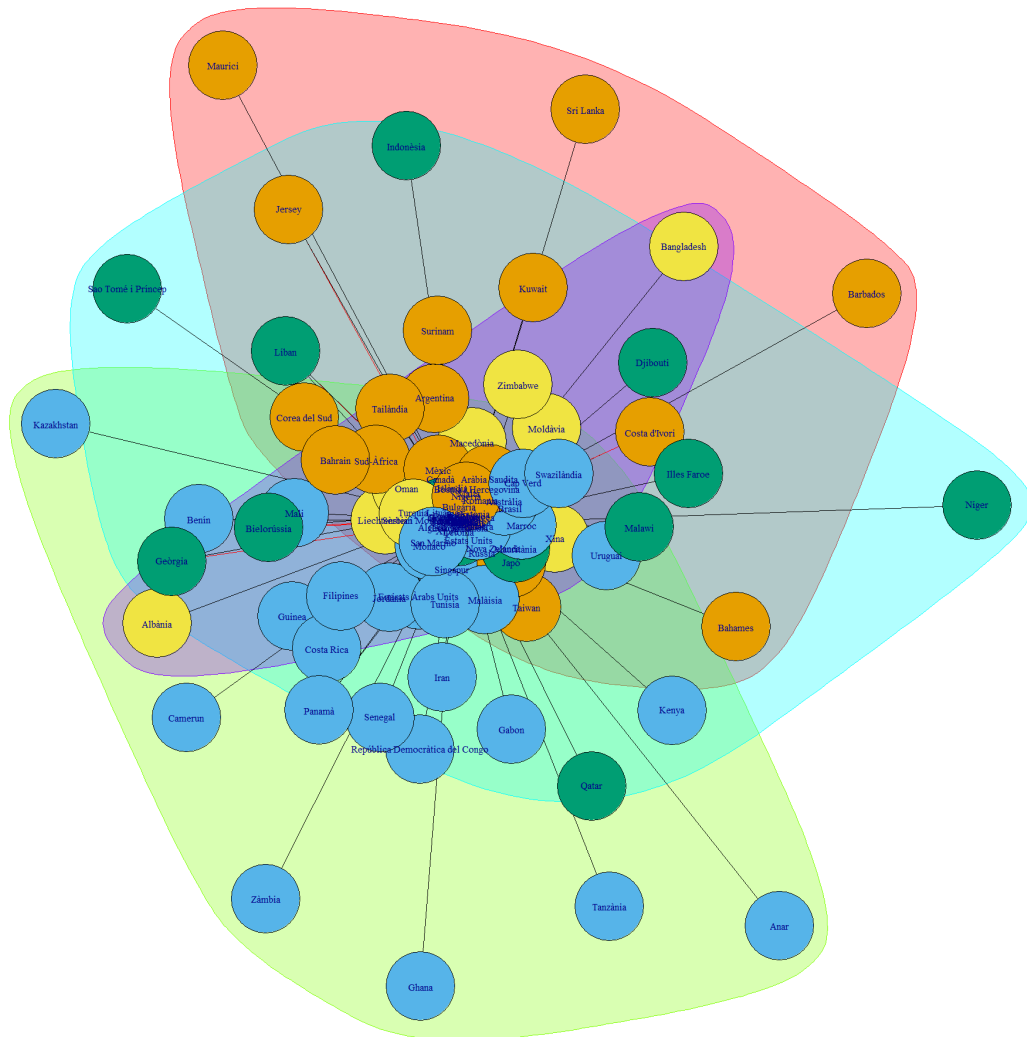


Figura 3.8: Representació visual de l'agrupació jeràrquica per clústers per la xarxa d'importacions des de tots els països de l'any 2018. El clúster de color vermell és l'1, el de color verd és el 2, el blau és el 3 i, finalment, el de color lila és el 4.

Tant si son aglomeratius com divisius, els mètodes jeràrquics produeixen tota una jerarquia de particions del graf i no només una única, que poden variar completament entre les dues particions: $\{\{v_1\}, \dots, \{v_{N_v}\}\}$ (si son aglomeratius) i V (si son divisius). La jerarquia resultats normalment es representa en forma d'arbre, anomenat **dendrograma**.

3.3.6 Assortativitat

Per enllestir l'anàlisi descriptiva de les característiques de les xarxes, s'exposarà un últim concepte: l'**assortativitat**.

Les mesures que quantifiquen l'extensió de la barreja assortativa en una xarxa determinada s'han denominat coeficients d'assortativitat i son essencialment variacions

del concepte de coeficient de correlació. Les característiques dels vèrtexs poden ser categòriques, ordinals o contínues. En el cas **categòric**, el coeficient és

$$r_a = \frac{\sum_i f_{ii} - \sum_i f_{i+} f_{+i}}{1 - \sum_i f_{i+} f_{+i}}, \quad (3.7)$$

on f_{ij} és la fracció d'enllaços de G que uneixen a un vèrtex en la categoria i amb un vèrtex de la categoria j , i f_{i+} i f_{+i} denoten la i -èssima fila i columna de sumes marginals, respectivament, de la matriu resultant \mathbf{f} . El valor r_a recau entre -1 i 1.

Quan les característiques son **contínues**, denotem per (x_e, y_e) els valors d'aquesta característica pels vèrtexs units per un enllaç $e \in E$ (en aquest cas, la població dels països). Aleshores, per quantificar l'assortativitat es pren el coeficient de correlació de Pearson pels parells (x_e, y_e) ,

$$r = \frac{\sum_{x,y} xy(f_{xy} - f_{x+} f_{+y})}{\sigma_x \sigma_y} \quad (3.8)$$

Així doncs, el coeficient d'assortativitat en la xarxa d'importacions des de tots els països a l'any 2018 és $r = -0.3390189$. Veiem que aquest valor és consistent amb l'anàlisi de la figura 3.5.

3.4 Anàlisi descriptiva II

En la secció anterior 3.3 s'ha desenvolupat l'anàlisi de les característiques de la xarxa d'importacions de residus des de tots els països a l'any 2018. Ara, per tal d'avaluar si hi ha diferències entre els residus, ens basarem en la variable del Conveni general de Basilea, la qual es troba descrita en la taula C.3. Com la variable imporant és la perillositat, primer es mirarà la distribució d'aquesta per veure si les dades estan balancejades o no. Tal i com es pot veure al gràfic 3.9(a), no ho estan perquè tant sols 4 de les 49 categories de la classificació de residus pertanyen a residus no perillosos. Així doncs, el 92% de les categories pertanyen a residus perillosos.

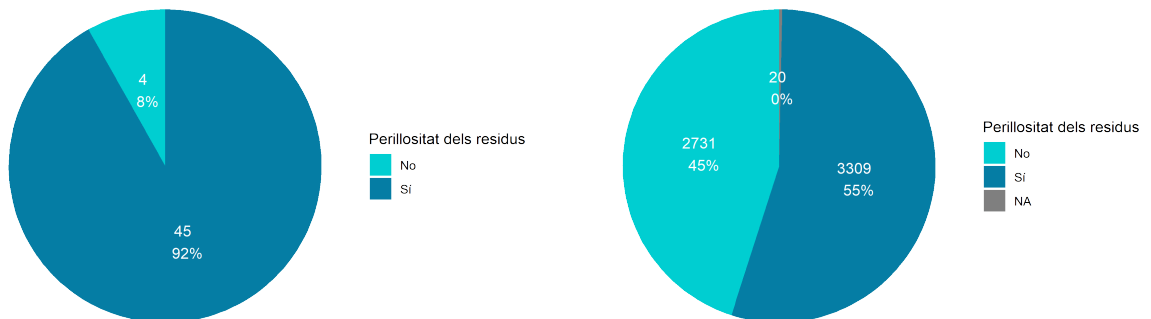


Figura 3.9: Distribució de les categories segons la variable perillositat dins la categoria del Conveni general de Basilea (a l'esquerra). Distribució dels residus segons la variable perillositat dins la categoria del Conveni general de Basilea filtrant per les importacions de l'any 2018.

A continuació es realitzarà el mateix procediment que abans, però només s'assenyalaran els resultats més rellevants per a l'anàlisi.

Començant per les **característiques dels nodes i els enllaços**, no s'hi presenten gaires diferències rellevants respecte els resultats obtinguts anteriorment. Els països que tenen major número de graus segueixen sent els mateixos (Alemanya, Bèlgica i França). Una característica que sí és remarcable és el número de nodes adjacents únics (veïns), doncs per als països que transporten residus perillosos només n'hi ha 5, mentre que pels que no ho son n'hi ha 19. A més a més, els vèrtexs de diferents graus estan connectats entre si de la mateixa manera que quan s'analitzava la xarxa global: hi ha tendència negativa, així que els nodes de major grau connecten amb els de menor, i viceversa (veure figures 3.10(a) i 3.10(b)).

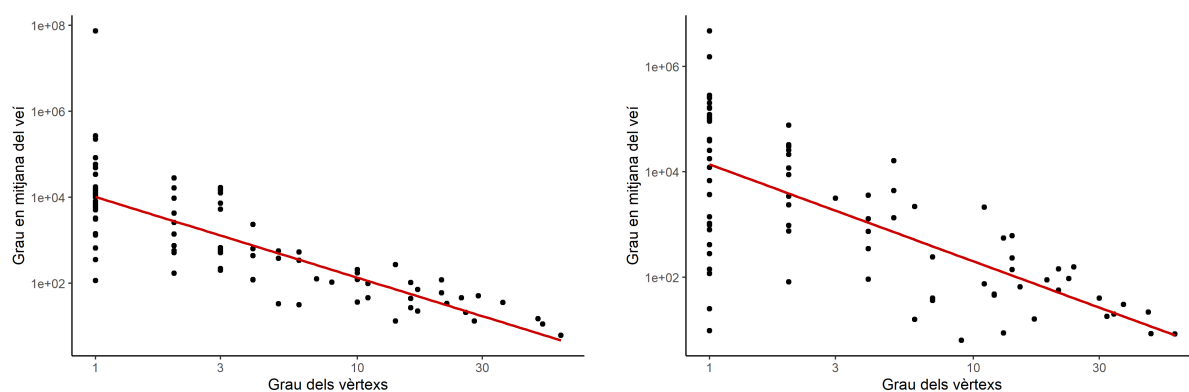


Figura 3.10: Grau mitja del veí en relació al grau dels nodes (en escala log-log) en importacions de residus perillosos (a l'esquerra) i no perillosos (a la dreta) des de tots els països l'any 2018.

Pel que fa a la centralitat dels nodes, s'obtenen els mateixos grafs que abans, però ara filtrant segons el tipus de perillositat dels residus (veure figures 3.11 i 3.12). Pels residus que sí son perillosos, no es veuen patrons destacables per la centralitat de proximitat dels vèrtexs amb un grau més elevat, i pels que no son perillosos tampoc. Pels residus perillosos no hi ha fets a destacar en quant a la interrelació, en canvi, pels que no ho son, els nodes grocs (graus elevats) estan força al centre, fet que indica que aquests nodes se situen en molts camins. Per últim, el graf de la centralitat del vector propi dels residus no perillosos indica que els països de França, Bèlgica i Alemanya son molt centrals en la xarxa i tenen més prestigi com a tal, per altra banda, pels residus perillosos no hi destaquen els nodes grocs.

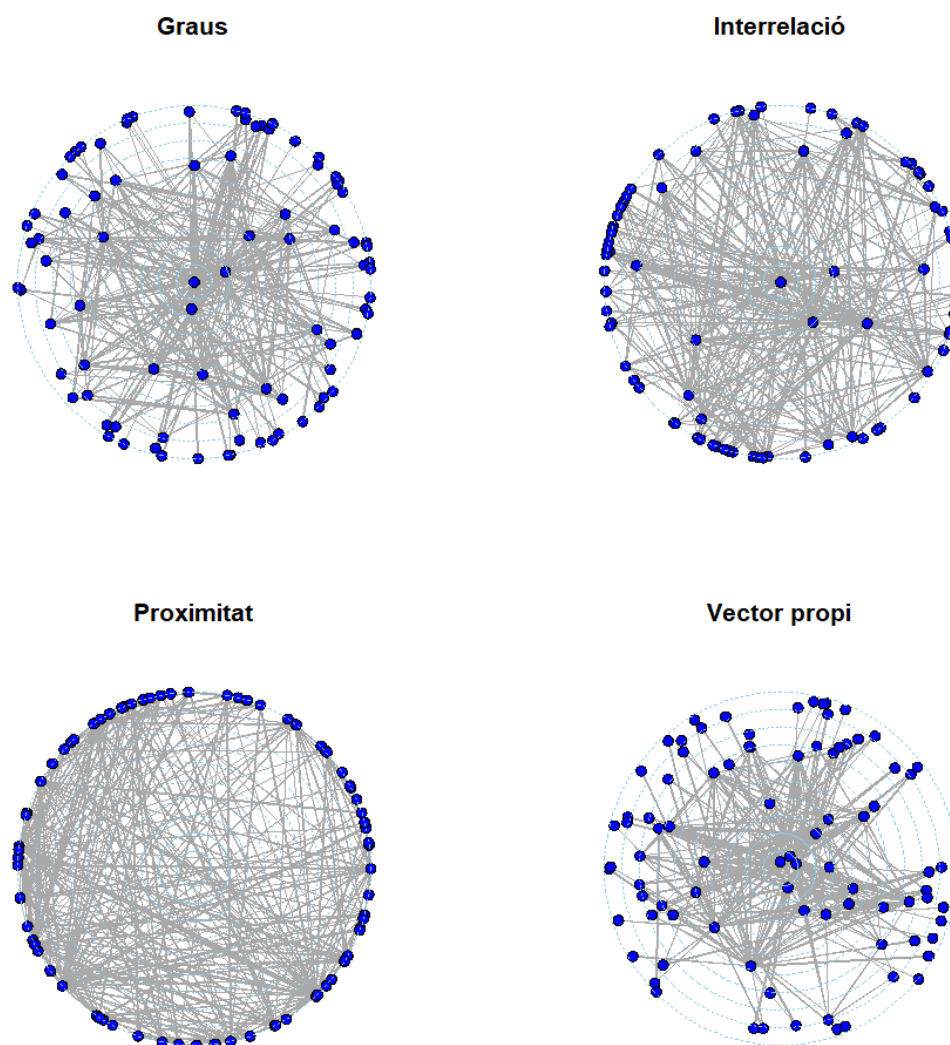


Figura 3.11: Grafts que mostren la centralitat dels nodes de la xarxa d'importacions del residus perillosos de tots els països l'any 2018. Els països amb un major grau, Alemanya, França i Bèlgica, es mostren en color groc, mentre que la resta de nodes estan pintats de blau.

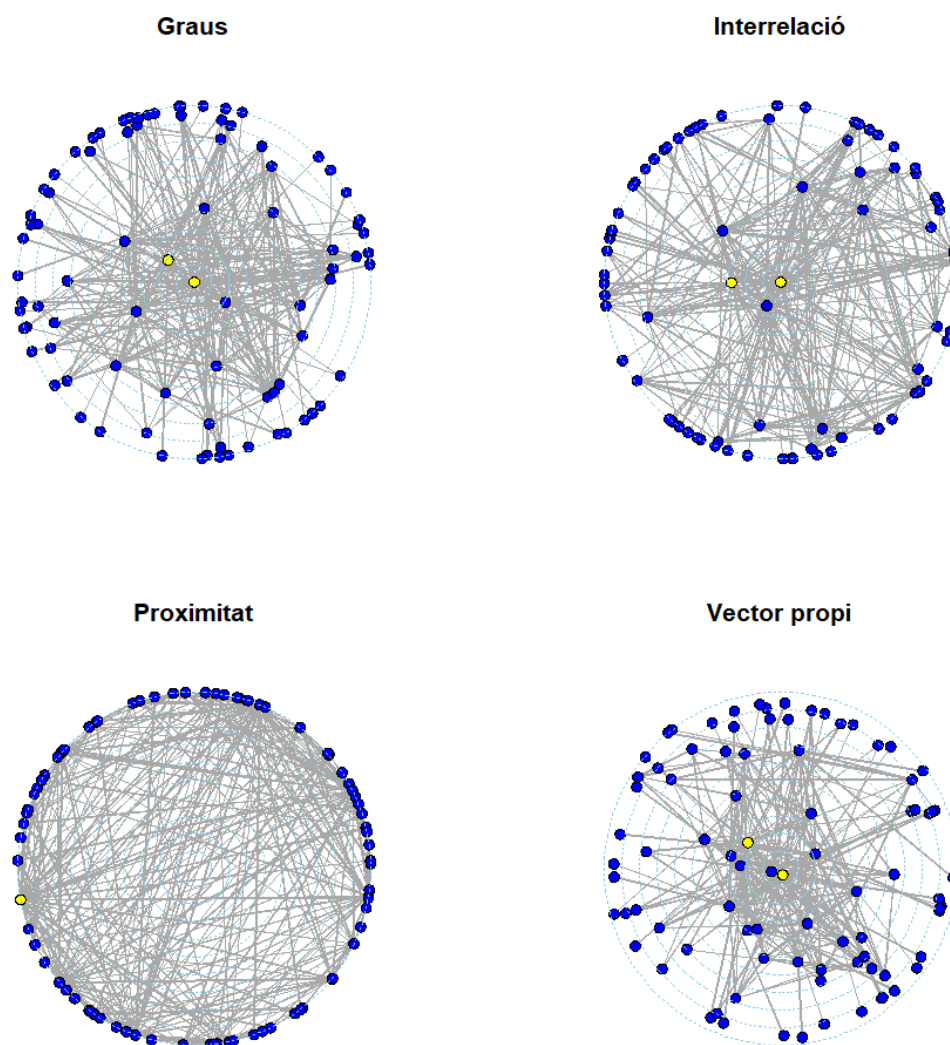


Figura 3.12: Grafs que mostren la centralitat dels nodes de la xarxa d'importacions del residu no perillós de tots els països l'any 2018. Els països amb un major grau, Alemanya, França i Bèlgica, es mostren en color groc, mentre que la resta de nodes estan pintats de blau.

En quant a la caracterització dels enllaços, els transportadors de residus perillós que tenen un paper més important en quant al fluxe de l'informació entre Alemanya i Suècia són el Regne Unit i Dinamarca. Pel que fa als no perillós, els que tenen un major pes són Alemanya i Àustria pel fluxe de Croàcia i Bèlgica.

Continuant per la **cohesió de la xarxa**, es mirarà quins són les cliques més grans en la taula 3.3. Es pot veure que tant pels residus perillós com pels que no ho són, la mida màxima del clique és de 9

Mida del clique	1	2	3	4	5	6	7	8	9
Freqüència absoluta (perillosos)	187	291	567	760	650	354	123	27	3
Freqüència absoluta del màxim (perillosos)	–	35	13	18	16	21	17	2	3
Freqüència absoluta (no perillosos)	86	270	512	684	613	380	161	42	5
Freqüència absoluta del màxim (no perillosos)	–	38	17	8	14	18	6	2	5

Taula 3.3: Freqüència absoluta censurant per cliques i pel màxim d'aquests per la xarxa d'importacions des de tots els països (classificats per perillositat) de l'any 2018.

En quant als nuclis, els nodes que tenen un k -core més gran amb $k = 321$ pels residus perillosos i $k = 227$ pels no perillosos corresponen als mateixos països, respectivament. són els mateixos que sense filtrar les dades en l'anàlisi global, és a dir: França, Països Baixos, Bèlgica i Alemanya. Hi ha una gran quantitat de díades nul·les, també.

Si mirem la densitat, gairebé no hi ha diferències entre les 2 xarxes donat que els valors són 0.04597701 per a la xarxa dels perillosos i 0.04637483 per a l'altra. La transitivitat tampoc ens aporta informació nova rellevant, tenen valors molt similars. La reciprocitat és major en els residus no perillosos que en els que sí que ho són, però els valors continuen essent baixos (hi ha un gran nombre de díades nul·les i poques mútues).

Si es fa un cop d'ull a l'agrupació jeràrquica, s'obté un resum per a cada clúster i per a cadascun dels datasets a les figures

El coeficient d'assortativitat en la xarxa d'importacions dels residus perillosos des de tots els països a l'any 2018 és $r = -0.3261359$ i pels no perillosos és $r = -0.3104508$. Veiem que aquest valor és consistent amb l'anàlisi de les figures 3.10(a) i 3.10(b).

Clúster	Mida (perillosos)	$\sum d_v$ (perillosos)	$\sum k$ (perillosos)	Mida (no perillosos)	$\sum d_v$ (no perillosos)	$\sum k$ (no perillosos)
1	31	1810	867	24	1194	821
2	20	2842	1527	31	1488	1046
3	26	1238	932	26	2551	1186
4	10 728	630	5	229	174	

Taula 3.4: Resum per a cada clúster segons el tipus de perillositat on es presenta: la seva mida, la suma grau dels nodes i la suma nucli dels nodes per a les regions de la xarxa d'importacions des de tots els països de l'any 2018.

I obtenim els clústers de les figures 3.13 i 3.14.

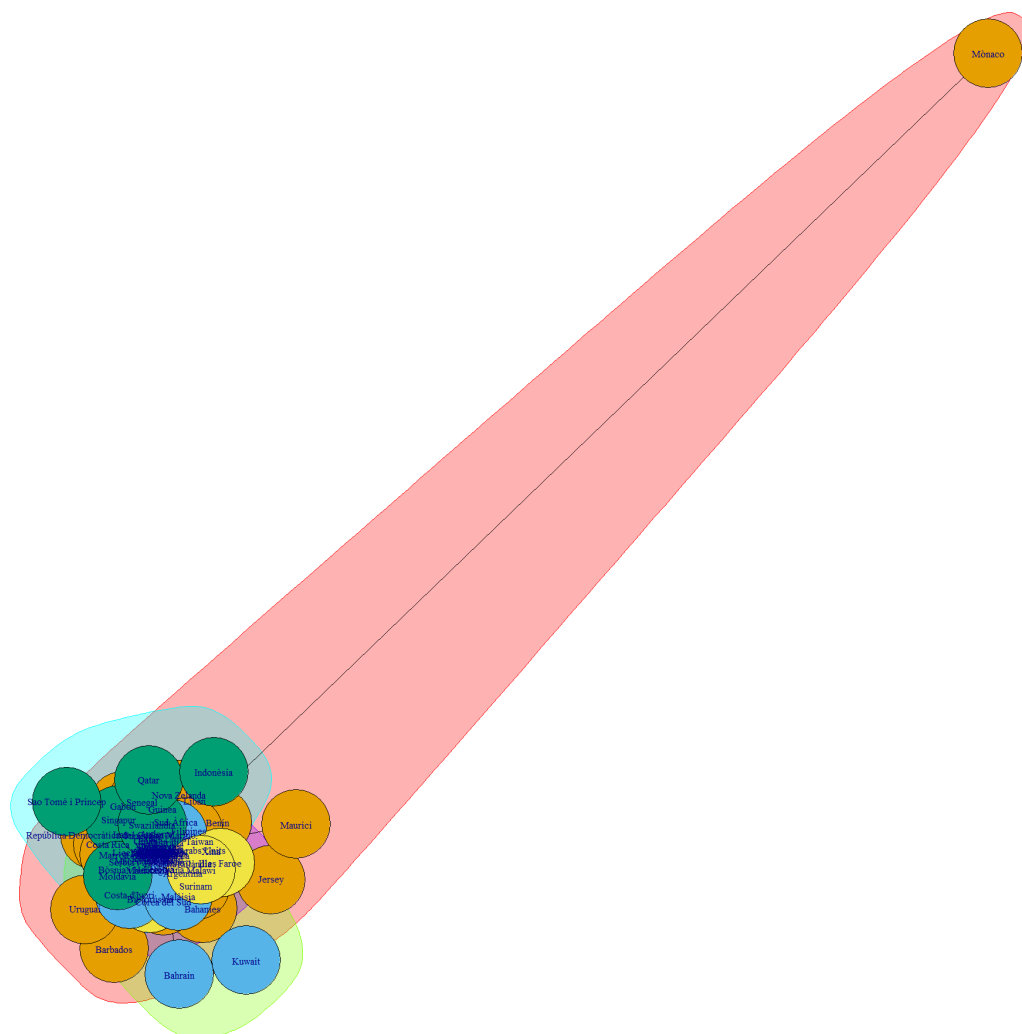


Figura 3.13: Representació visual de l'agrupació jeràrquica per clústers per la xarxa d'importacions de residus no perillosos des de tots els països de l'any 2018. El clúster de color vermell és l'1, el de color verd és el 2, el blau és el 3 i, finalment, el de color lila és el 4.

	Perillositat	No perillositat	Global
N_v	87	86	180
N_e	3309	2731	6060
d_v	Alemanya, Bèlgica, França	Alemanya, Bèlgica, França	Alemanya, Bèlgica, França
d_v^{in}	Alemanya, Bèlgica, França	Alemanya, Bèlgica, França	Alemanya, Bèlgica, França
d_v^{out}	Països Baixos, Itàlia, França	Països Baixos, Itàlia, França	Països Baixos, Itàlia, França
Nodes adjacents únics	5	19	20
Centralitat nodes	Relació negativa	Relació negativa	Relació negativa
k-cores	321	227	597
Díades nul·les	3450	3385	5392
Densitat	0.04597701	0.04637483	0.04153686
Transitivitat	0.3125115	0.3327556	0.3275322
Reciprocitat (enllaços dirigits)	0.1821306	0.2555556	0.2435233
Assortativitat	-0.3261359	-0.3104508	-0.3390189

Mitjançant l'app web Shiny, podem veure els països que tenen un número de graus majors, així com també un major número de nuclis, a les figures 3.15 i 3.16. En aquests, es pot veure que pels països amb un major número de graus i nuclis, Alemanya, França i Bèlgica, importen (i també exporten) grans quantitats de residus. En el cas d'Alemanya, l'any 2018 va importar 2100811.67 tones de residus però també va exportar 3268986.13 tones de residus, principalment en països d'Europa en ambdós casos. En el segon graf, es pot veure que hi ha un gran fluxe de transport de residus no perillosos entre Itàlia i França, i entre Dinamarca, República Txeca, Polònia i Àustria cap Alemanya.

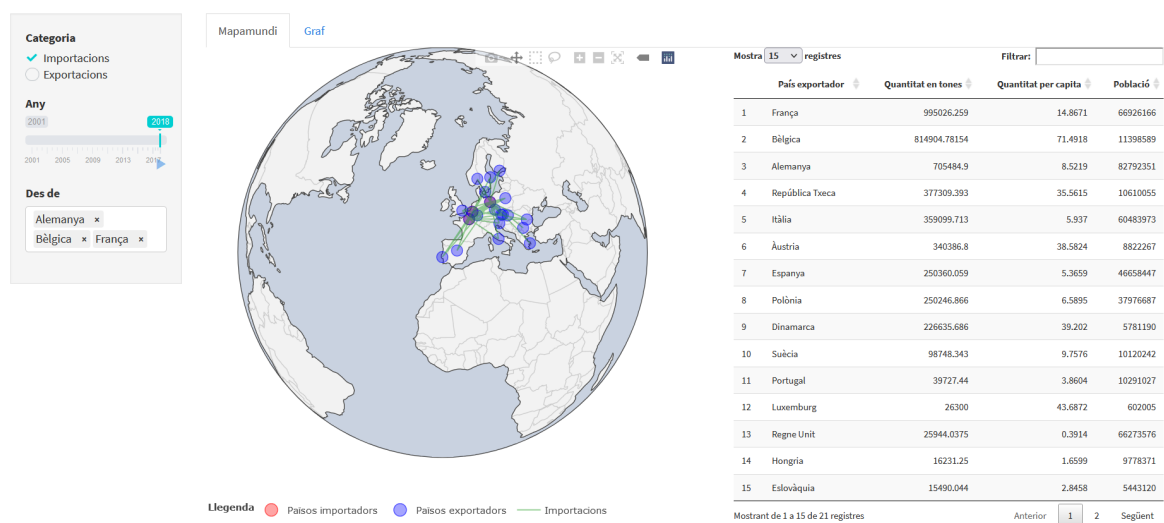


Figura 3.15: Captura de pantalla de l'aplicació Shiny on es mostra el mapamundi de les importacions des d'Alemanya, França i Bèlgica de l'any 2018

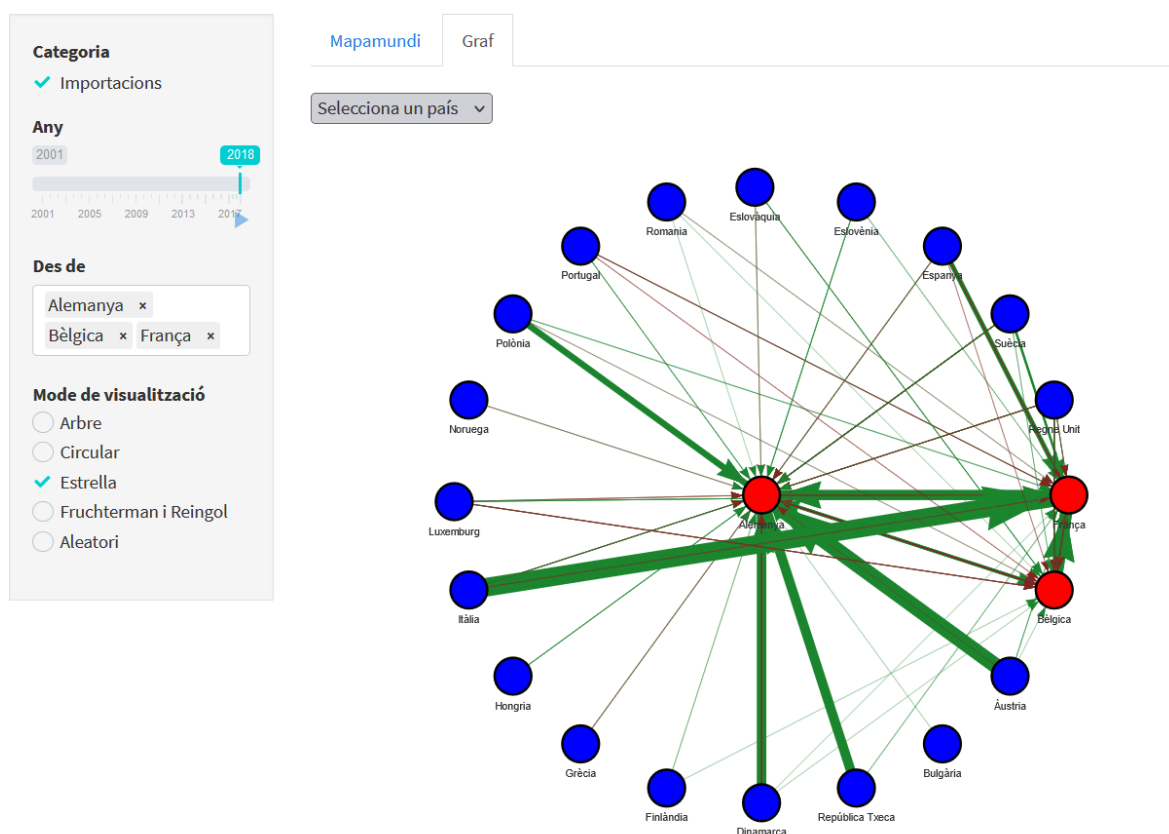


Figura 3.16: Captura de pantalla de l'aplicació Shiny on es mostra el graf de les importacions des d'Espanya i Portugal en disposició d'arbre l'any 2018

4 | Conclusions

Per finalitzar el treball, es farà un recull dels resultats més importants que donaran peu a respondre als objectius plantejats inicialment, que principalment eren els següents: observar les relacions que hi ha entre les importacions de residus dels Estats Membres de la UE i agregats de la UE cap a països socis, que no tenen per què ser només Estats Membres de la UE; i veure si hi havia diferències entre els residus classificats com a perillosos o no segons el Conveni general de Basilea.

En primer lloc, s'ha pogut veure que no existeixen grans diferències entre les xarxes dels residus perillosos en relació amb la xarxa global, així com tampoc n'hi ha pels residus classificats com a no perillosos. És a dir, les tres xarxes tenen una densitat semblant, malgrat que la xarxa de residus no perillosos la té més elevada que les altres dues. Per tant, aquesta xarxa és la que més cliques (subgrafs complets, és a dir, que cada vèrtex està unit a tots els altres per un enllaç) té, en comparació de les altres dues. Pel que fa a la connectivitat de les xarxes, hi ha una tendència negativa per cadascuna de les xarxes que es pot explicar de la mateixa manera: els nodes amb major grau es relacionen amb els de menor grau, i viceversa. En l'actualitat, no és cap sorpresa escoltar o llegir notícies (com la de [“Los países de la UE exportan unas 352.500 toneladas al año de residuos electrónicos a países en desarrollo” 2019](#) que expliquen que les potències econòmiques envien residus, en aquest cas d'aparells electrònics, cap als països en vies de desenvolupament. Si per exemple es filtren les importacions pel país de Nigèria a l'any 2017 a l'app web, que és un dels països de destinació de residus que l'article indica, es pot comprobar que els països que exporten residus cap allà són Alemanya, Polònia, Grècia i Itàlia, ordenats de major a menor quantitat de residus enviats.

Una de les grans limitacions d'aquest treball és la inconsistència que hi ha entre importacions i exportacions a la base de dades, ja que quan l'enviament es fa entre dos Estats Membres de la UE, en principi s'hauria de poder mostrar la mateixa informació sobre allò que els països exportadors i importadors han informat, però no és així. Per tant, això limita en quant a l'obtenció de resultats acurats i precisos, ja que mai se sabrà si les dades reportades pels països són correctes i no tenen cap mena d'error. En l'exemple anterior, l'any 2017 Nigèria va informar de l'importació de residus procedents d'Alemanya, Polònia, Grècia i Itàlia, però Alemanya no va reportar el seu enviament de residus (és a dir, exportacions) cap a Nigèria. És el que es comenta al principi del document, que en general hi ha un major número de registres reportats en importacions que no pas d'exportacions.

Per últim, m'agradaria donar una breu valoració personal del treball i de futures tasques que crec que es podrien dur a terme mitjançant la teoria de grafs. Malgrat no haver aprofundit per a totes les possibles casuístiques considerant totes les variables, crec que seria oportú realitzar-ho de cara a una anàlisi més extensa i, alhora, s'hi podria

implementar a l'app web, de la mateixa manera que també s'hi podrien implementar altres funcions. De totes formes, crec que aquesta pot ser una eina molt útil de cara a l'extracció de resultats claus per a la conscienciació de la societat (tant a nivell individual com a nivell col·lectiu d'empreses, organitzacions, institucions, etc.) sobre la generació de residus. Sovint, ni nosaltres mateixos som conscients de molts tipus de dades que ens envolten fins que no els veiem representats sobre un gràfic, per exemple, per això és molt important aprendre a gestionar les dades correctament mitjançant eines potents, com bé pot ser l'anàlisi de xarxes que, mitjançant una imatge, permeten generar idees sense necessitat de ser coneixedors de conceptes o fórmules matemàtiques.

5 | Referències

- Brandes, Ulrik i Thomas Erlebach (2005). *Network Analysis. Methodological Foundations*. Springer, Berlin, Heidelberg. DOI: <https://doi.org/10.1007/b106453>.
- Climate change – Overview*. Eurostat. URL: <https://ec.europa.eu/eurostat/web/climate-change/overview> (cons. 25-02-2021).
- Gábor, Csárdi (2021). *transitivity: Transitivity of a graph*. URL: <https://rdr.io/cran/igraph/man/transitivity.html> (cons. 06-04-2021).
- Kolaczyk, D. Eric. i Csárdi Gábor (2020). *Statistical Analysis of Network Data with R*. Second. Springer. DOI: <https://doi.org/10.1007/978-3-030-44129-6>.
- Lizardo, Omar i Isaac Jilbert (6 de gen. de 2020). *Social Networks: An Introduction. Density*. URL: https://bookdown.org/omarlizardo/_main/2-9-density.html (cons. 07-07-2021).
- “Los países de la UE exportan unas 352.500 toneladas al año de residuos electrónicos a países en desarrollo” (1 de març de 2019). A: *europapress*. URL: <https://www.europapress.es/sociedad/medio-ambiente-00647/noticia-paises-ue-exportan-352500-toneladas-ano-residuos-electronicos-paises-desarrollo-20190301164323.html> (cons. 20-07-2021).
- Plotly R Open Source Graphing Library* (2021). plotly. URL: <https://plotly.com/r/> (cons. 28-05-2021).
- ¿Qué es el cambio climático?* Acciona. URL: <https://www.acciona.com/es/cambio-climatico/>.
- Shiny* (2020). RStudio. URL: <https://shiny.rstudio.com/> (cons. 18-07-2021).
- Sievert, Carson (19 de des. de 2019). *Interactive web-based data visualization with R, plotly, and shiny. Maps*. URL: <https://plotly-r.com/maps.html> (cons. 28-05-2021).
- visNetwork, an R package for interactive network visualization* (2017). DataStorm. URL: <https://datastorm-open.github.io/visNetwork/> (cons. 07-06-2021).
- Waste shipments across borders* (18 de nov. de 2019). Eurostat. URL: <https://ec.europa.eu/eurostat/web/waste/data> (cons. 04-03-2021).
- Who we are*. Eurostat. URL: <https://ec.europa.eu/eurostat/about/who-we-are> (cons. 25-02-2021).
- Who we are* (21 de febr. de 2020). URL: <https://www.oecd.org/about/> (cons. 26-06-2021).
- Wikipedia (2021a). *Assortativity*. URL: <https://en.wikipedia.org/wiki/Assortativity> (cons. 06-04-2021).
- (2021b). *Netherlands Antilles – Wikipedia, The Free Encyclopedia*. URL: https://en.wikipedia.org/w/index.php?title=Netherlands_Antilles&oldid=1029678793 (cons. 25-04-2021).

- Wikipedia (2021c). *North Macedonia – Wikipedia, The Free Encyclopedia*. URL: https://en.wikipedia.org/w/index.php?title=North_Macedonia&oldid=1032654005.
- (2021d). *Sèrbia – Viquipèdia, l'Enciclopèdia Lliure*. URL: <https://ca.wikipedia.org/w/index.php?title=S%C3%A8rbia&oldid=27680870>.

Annexos

A | Base de dades

Per altra banda, en la fulla de l'explicació de les variables (***Explanation of columns***) s'hi troba l'explicació detallada per a cadascuna de les variables del conjunt de dades que apareix a la primera fulla de càlcul tal que així:

Year És l'any de referència de les dades. La sèrie de dades comença en el 2001 i acaba en el 2018. No obstant això, els països només han estat obligats a informar de dades quan els Estats Membres de la UE ho havien fet. Com a tal, per a alguns països, el primer any de referència és posterior al 2001.

Import/export Indica si les dades son importacions o exportacions.

Country reporting País que reporta. *Nota:* Per exemple, si es selecciona Àustria i “exportar” a la variable *Import/export*, les variables següents mostren el que ha informat Àustria sobre les quantitats exportades. Quan l'enviament es faci entre dos Estats Membres de la UE, en principi s'hauria de poder mostrar informació sobre allò que els països exportadors i importadors han informat sobre el mateix enviament de residus. Tanmateix, hi ha diferències en la presentació d'informes entre els països respectius (per exemple, l'ús de diferents codis de classificació o tractament de residus, referència a l'aplicació (volum i any previst) o l'enviament efectiu (volum i any quan es produeix), o que l'Estat Membre exportador ha inclòs els residus de la llista verda notificats als seus informes, mentre que l'Estat Membre receptor no) que volen dir que a la pràctica hi poden haver algunes discrepàncies.

Population Població del país que reporta.

Quantity in tonnes Quantitat d'enviament de residus en tones.

Quantity in kg per capita Quantitat d'enviament de residus en kg per capita.

To or from country Quan l'enviament de residus que es descriu és una exportació, el país d'aquesta variable és el país receptor. En canvi, quan l'enviament de residus és una importació, el país que representa aquesta variable és el país d'enviament. *Nota:* Si es filtra per aquesta columna es mostraran tots els residus reportats importats i exportats a aquest país per la resta de països. S'ha de tenir en compte que un país pot exportar residus o importar-ne mitjançant el trànsit a través d'un tercer país. Per exemple, Àustria envia residus d'una regió del país a una altre a través d'Alemanya. En aquest cas, Àustria és el país exportador i importador.

Disposal and recovery code És la operació de tractament per a la qual s'envien els residus. Els codis s'enumeren a l'annex I i II de la Directiva Marc de residus de la UE 2008/98/EC. 'R' significa que els residus son per a una operació de recuperació i 'D' significa que els residus son per a una operació d'eliminació. Es podia indicar més d'un codi d'operació quan els residus enviats rebien més d'un tipus de tractament.

General Basel Convention code -Y code Indica el tipus de residu segons l'annex I i II del Conveni de Basilea. A l'annex I s'inclouen els residus (codis Y1-Y45) que son subjectes de moviments transfronterers i que es consideren perillosos als efectes del Conveni. A l'annex II s'inclouen altres residus (codis Y46-Y47) que no es consideren perillosos als efectes del Conveni, però que encara requereixen una consideració especial.

European List of Waste code És el tipus de residu segons la Llista Europea de Residus. Els codis s'enumeren a l'annex de la Decisió 2014/955/EU que modifica la Decisió 200/532/EC. Els codis amb un * indiquen que els residus es consideren perillosos. Alguns tipus de residus no inclouen un *, però han de ser notificats segons la legislació de la UE o la decisió de l'OCDE.

Detailed Basel Convention code or OECD decision code Mostra informació detallada sobre els residus enviats segons el Conveni de Basilea i la Decisió C(2001)107. L'annex VIII (llista A) del Conveni de Basilea proporciona un codi detallat per als residus perillosos, mentre que l'annex IX (llista B) del Conveni proporciona una visió general dels codis de residus no perillosos. L'annex IV, part II de la Decisió de l'OECD, inclou codis addicionals a l'annex VIII i l'annex IX del Conveni de Basilea.

Basis for classification to hazardous and non-hazardous by CRI Indica la classificació dels residus com a perillosos o no perillosos juntament amb el motiu de la classificació. Aquesta classificació la va realitzar el CRI. La base per classificar-los com a perillosos pot ser que a) els residus es reporten com un codi Y entre Y1-Y45, o b) un codi de la Llista de Residus Europea amb un * indica que els residus son perillosos, o c) el país que informa ha indicat a la informació que els residus son perillosos tot i estar vinculats a un codi de residus no perillosos.

UN hazardous class list L'annex III del Conveni de Basilea inclou una llista de característiques perilloses segons la classificació de les Nacions Unides. L'annex III inclou informació sobre la classe, el codi i les característiques de les Nacions Unides. La variable *Detailed Basel Convention code or OECD decision code* indica la classe de les Nacions Unides i la variable *Basis for classification to hazardous and non-hazardous by CRI* indica les propietats del codi perillós (el codi de *Disposal and recovery code*), que fan perillosos els residus. Aquesta llista correspon al sistema de classificació de perills inclòs a les Recomanacions de les Nacions Unides sobre el transport de mercaderies perilloses (ST/SG/AC.10/1Rev.5, Nacions Unides, Nova York, 1988).

Country of transit stated by a code País o països pels quals han passat els residus transfronterers enviats durant l'enviament del país exportador al país impor-

tador. El país o els països de trànsit s'indiquen mitjançant un codi de país de dues lletres segons la norma ISO 3166.

Notes son notes addicionals sobre l'enviament proporcionades pel país reportant. Pot ser una descripció dels residus i la base per classificar-los com a perillosos. Per exemple, sovint el país informant declara "1/1(b)", que significa que els residus es classifiquen com a perillosos segons l'article 1(1)b del Conveni de Basilea. És a dir, perillos segons la classificació de la UE o classificació nacional, però no perillós segons l'article 1(1)a del Conveni de Basilea. Les notes també poden incloure un codi de la llista europea de residus (*Disposal and recovery code*), un codi detallat del Conveni de Basilea o un codi de decisió de l'OCDE (*General Basel Convention code -Y code*) o una llista de classes perilloses de l'ONU (*Detailed Basel Convention code or OECD decision code* i *Basis for classification to hazardous and non-hazardous by CRI*). L'ideal seria que aquesta informació s'hagi de transportar a la columna corresponent per obtenir aquestes dades. Tanmateix, això requeriria una transposició manual de fins a 280000 línies de dades, i s'ha considerat massa prohibitiu a nivell de recursos.

B | Preprocessament de les dades

Els canvis realitzats per unificar els noms de tots els països del món amb els de la base de dades **waste** es troben aquí:

- Totes les regions (és a dir, països) del dataset **world** que contenen Antigua o Barbuda passen a anomenar-se “Antigua i Barbuda”.
- Totes les regions (és a dir, països) del dataset **world** que contenen Trinidad o Tobago passen a anomenar-se “Trinidad i Tobago”.
- Totes les regions (és a dir, països) del dataset **world** que contenen UK passen a anomenar-se “United Kingdom”.
- Totes les regions (és a dir, països) del dataset **world** que contenen USA passen a anomenar-se “United States”.
- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **world** que contenen Brunei passen a anomenar-se “Brunei”. Això és degut a que en **waste** aquest país apareix com a “Brunei Darussalam”.
- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **waste** que contenen China passen a anomenar-se “China”. Això és degut a que en **waste** aquest país apareix com a “China (including Hong Kong)”.
- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **waste** que contenen Congo passen a anomenar-se “Democratic Republic of the Congo”. Això és degut a que en **waste** aquest país apareix com a “Congo” i com a “Democratic Republic of the Congo”. Cal tenir en compte que existeixen dos països amb aquest nom: la República del Congo i la República Democràtica del Congo, per ser consistents en aquest estudi triarem el segon país.
- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **waste** que contenen Côte passen a anomenar-se “Ivory Coast”. Això és degut a que en **waste** aquest país apareix com a “Côte d’Ivoire”.
- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **waste** que contenen Czechia passen a anomenar-se “Czech Republic”. Això és degut a que en **waste** aquest país apareix com a “Czechia”.
- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **waste** que contenen Antilles passen a anomenar-se “Netherlands”. Això és degut a que en **waste** aquest país apareix com a “Former Netherlands Antilles”, que va ser un conjunt d’illes dissoltes el 2010 (Wikipedia, [2021b](#)).

- Tots els països de les variables *Country.reporting* o *To.or.from.country* del dataset **waste** que contenen Sao Tome And Principe passen a anomenar-se “Sao Tome and Principe”.

Els canvis realitzats per unificar els noms de tots els països del món amb els de les bases de dades **waste** i **capitals** es troben a continuació:

- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Pitcairn passen a anomenar-se “Pitcairn Islands”. Això és degut a que en **world.cities** aquest país apareix com a “Pitcairn”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Kitts passen a anomenar-se “Saint Kitts”. Això és degut a que en **world.cities** aquest país apareix com a “Saint Kitts and Nevis”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Congo passen a anomenar-se “Democratic Republic of the Congo”. Això és degut a que en **world.cities** aquest país apareix com a “Congo” i com a “Congo Democratic Republic”. De la mateixa forma que abans, per ser consistents en aquest estudi triarem el segon país.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Turks passen a anomenar-se “Turks and Caicos Islands”. Això és degut a que en **world.cities** aquest país apareix com a “Turks and Caicos”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Timor passen a anomenar-se “Timor-Leste”. Això és degut a que en **world.cities** aquest país apareix com a “East Timor”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Vincent passen a anomenar-se “Saint Vincent”. Això és degut a que en **world.cities** aquest país apareix com a “Saint Vincent and The Grenadines”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen UK passen a anomenar-se “United Kingdom”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Korea North passen a anomenar-se “North Korea”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Guernsey passen a anomenar-se “Guernsey”. Això és degut a que en **world.cities** aquest país apareix com a “Guernsey and Alderney”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Korea South passen a anomenar-se “South Korea”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen Vatican passen a anomenar-se “Vatican”. Això és degut a que en **world.cities** aquest país apareix com a “Vatican City”.
- Tots els països de la variable *country.etc* del dataset **world.cities** que contenen USA passen a anomenar-se “United States”.

- Tots els països de la variable *country.etc* del dataset `world.cities` que contenen Antilles passen a anomenar-se “Netherlands”. Això és degut a que en `world.cities` aquest país apareix com a “Netherlands Antilles”, que va ser un conjunt d’illes dissoltes el 2010 (Wikipedia, [2021b](#)).

Pel país Kosovo no hi ha registre en el dataset de `world.cities`; en canvi, la capital de Kosovo (Pristina) sí que apareix però consta que el país és Sèrbia i Montenegro. Així doncs, com es tracta d’un error de la base de dades, s’han d’introduir a mà les coordenades per a Kosovo.

```

1 # Paquets necessaris
2 library(tidyverse)
3 library(xlsx)
4 library(readxl)
5 library(maps)
6
7 #####
8 ### World ###
9 #####
10
11 world <- map_data("world") %>%
12   # Es corregeixen i/o s'actualitzen els noms dels països
13   mutate_at(.vars = vars(region),
14     .funs = list(~ case_when(
15       str_detect(., pattern = "Antigua") ~ "Antigua and
16         Barbuda",
17       str_detect(., pattern = "Barbuda") ~ "Antigua and
18         Barbuda",
19       str_detect(., pattern = "Trinidad|Tobago") ~ "Trinidad
20         and Tobago",
21       str_detect(., pattern = "UK") ~ "United Kingdom",
22       str_detect(., pattern = "USA") ~ "United States",
23       str_detect(., pattern = "Serbia|Montenegro") ~ "Serbia
24         and Montenegro",
25       TRUE ~ .
26     ))) %>%
27   # Es converteixen a classe factor algunes variables
28   mutate_at(.vars = vars(region),
29     .funs = list(factor)) %>%
30   select(-subregion)
31
32 #####
33 ### Capitals ###
34 #####
35
36 # Importació de les dades
37 data("world.cities")
38
39 capitals <- world.cities %>% filter(capital == 1) %>%
40   # Es corregeixen i/o s'actualitzen els noms dels països
41   mutate_at(.vars = vars(country.etc),
42     .funs = list(~ case_when(
43       str_detect(., pattern = "Pitcairn") ~ "Pitcairn Islands"
44       ,
45       str_detect(., pattern = "Kitts") ~ "Saint Kitts",
46       str_detect(., pattern = "Congo") ~ "Democratic Republic
47         of the Congo", # assumim que és de la República
48         Democràtica del Congo
49       str_detect(., pattern = "Turks") ~ "Turks and Caicos
50         Islands",
51       str_detect(., pattern = "Timor") ~ "Timor-Leste",
52       str_detect(., pattern = "Vincent") ~ "Saint Vincent",
53       str_detect(., pattern = "UK") ~ "United Kingdom",
54       str_detect(., pattern = "Korea North") ~ "North Korea",
55       str_detect(., pattern = "Guernsey") ~ "Guernsey",
56       str_detect(., pattern = "Korea South") ~ "South Korea",

```

```

50         str_detect(., pattern = "Vatican") ~ "Vatican",
51         str_detect(., pattern = "USA") ~ "United States",
52         str_detect(., pattern = "Antilles") ~ "Netherlands",
53         TRUE ~ .
54     ))) %>%
55     select(name, country.etc, lat, long) %>%
56     # Afegim Kosovo
57     add_row(name = "Pristina", country.etc = "Kosovo", lat = 42.65, long
58             = 21.17) %>%
59     rename(capital = name,
60            country = country.etc) %>%
61     # Es converteixen a classe factor algunes variables
62     mutate_at(.vars = vars(capital,
63                             country),
64               .funs = list(factor)) %>%
65     # Hi ha 5 països duplicats, així que es selecciona el primer cas de
66     # cadascun dels països
67     group_by(country) %>%
68     top_n(n = 1)
69 #####
70 ### Waste ###
71 #####
72
73 # Importació de les dades
74 waste <- read_csv("../data/waste.csv", header = TRUE, sep=";", check.
75                  names = FALSE, dec = ",",) %>%
76     tibble(.name_repair = "universal")
77
78 # Traducció de les categories de la variable Codi d'eliminació i
79 # recuperació
80 eu_wasteI <- read_excel("../data/codification/codification_eu_waste_
81                        trans.xlsx", sheet = "I_disposal_operations")
82 eu_wasteII <- read_excel("../data/codification/codification_eu_waste_
83                        trans.xlsx", sheet = "II_recovery_operations")
84 eu_waste <- rbind(eu_wasteI, eu_wasteII)
85
86 # Traducció de les categories de la variable Conveni general de
87 # Basilea
88 basel_convI.1 <- read_excel("../data/codification/codification_basel_
89                        convention_trans.xlsx", sheet = "I_waste_streams")
90 basel_convI.2 <- read_excel("../data/codification/codification_basel_
91                        convention_trans.xlsx", sheet = "I_waste_having_as_constituents")
92 basel_convII <- read_excel("../data/codification/codification_basel_
93                        convention_trans.xlsx", sheet = "II_requiring_special_consideration")
94 basel_conv <- rbind(basel_convI.1, basel_convI.2, basel_convII)
95
96 # Funció per a recodificar NAs
97 replace_factor_na <- function(x){
98     x <- as.character(x)
99     x <- if_else(is.na(x), "Sense descripció", x)
100    x <- as.factor(x)
101 }

```



```

97 waste <- waste %>%
98 # Es corregeixen i/o s'actualitzen els noms dels països
99 mutate_at(.vars = vars(Country.reporting, To.or.from.country),
100 .funs = list(~ case_when(
101   str_detect(., pattern = "Brunei") ~ "Brunei",
102   str_detect(., pattern = "China") ~ "China",
103   str_detect(., pattern = "Congo") ~ "Democratic Republic
    of the Congo",
104   str_detect(., pattern = "Côte") ~ "Ivory Coast",
105   str_detect(., pattern = "Czechia") ~ "Czech Republic",
106   str_detect(., pattern = "Macedonia") ~ "Macedonia",
107   str_detect(., pattern = "Antilles") ~ "Netherlands",
108   str_detect(., pattern = "Serbia|Montenegro") ~ "Serbia
    and Montenegro",
109   str_detect(., pattern = "Sao Tome And Principe") ~ "Sao
    Tome and Principe",
110   str_detect(., pattern = "World") ~ "Not specified",
111   TRUE ~ .
112 ))) %>%
113 # Es recodifiquen les categories de la variable d"importacions/
    exportacions
114 mutate_at(.vars = vars(Import.export),
115   recode, "import" = "Import", "export" = "Export") %>%
116 # Es converteixen a classe factor algunes variables
117 mutate_at(.vars = vars(Import.export,
118   Country.reporting,
119   To.or.from.country,
120   Disposal.and.recovery.code,
121   General.Basel.Convention.code..Y.code,
122   European.List.of.Waste.code,
123   Detailed.Basel.Convention.code.or.OECD.
    decison.code,
124   Basis.for.classification.to.hazardous.and.non
    .hazardous,
125   UN.hazardous.code.properties.which.render.the
    .waste.hazardous),
126 .funs = list(factor)) %>%
127 # S'arregla i s'assignen etiquetes a la variable Codi d'eliminació i
    recuperació
128 mutate_at(.vars = vars(Disposal.and.recovery.code),
129   recode, "D9 " = "D9") %>%
130 mutate_at(.vars = vars(Disposal.and.recovery.code),
131   .funs = list(~ factor(., levels = eu_waste$code, labels =
    eu_waste$description))) %>%
132 # S'arregla i s'assignen etiquetes a la variable Conveni general de
    Basilea
133 mutate_at(.vars = vars(General.Basel.Convention.code..Y.code),
134   .funs = list(~ factor(., levels = basel_conv$code, labels
    = basel_conv$description))) %>%
135 # S'afegeix la variable de perillositat
136 mutate(hazardous = )
137 # Es recodifiquen els NAs que falten
138 mutate_if(is.factor, replace_factor_na)
139
140
141 # S'extrauen els noms dels països de world
142 world.countries <- data.frame(region = unique(world$region))

```

```

143
144 # S'extrauen els noms dels països de waste
145 waste.countries <- data.frame(country = union(waste$Country.reporting,
146       waste$To.or.from.country))
147
148 # Es miren quins són els noms dels països que estan en waste però no
149 # en world
150 anti_join(x = waste.countries, y = world.countries, by = c("country" =
151       "region"))
152
153 # Es miren quins són els noms dels països que estan en capitals però
154 # no en world
155 anti_join(x = capitals, y = world.countries, by = c("country" = "
156       region")) # British Virgin Islands, US Virgin Islands, Tokelau,
157       Gibraltar, Svalbard and Jan Mayen, Tuvalu: no existeixen en world
158       (no són països com a tal)
159
160 # Coordenades per país
161 world_coord <- left_join(x = waste.countries, y = capitals, by = "
162       country") %>%
163       select(-capital) %>%
164       # Es converteixen a classe factor algunes variables
165       mutate_at(.vars = vars(country),
166             .funs = list(factor))
167
168 # S'assignen les coordenades 0,0 a la categoria "Not specified"
169 world_coord[world_coord$country == "Not specified",][,c(2, 3)] <- 0
170
171 # Es guarden els noms dels països en fitxers Excel
172 # write.xlsx(levels(world$region), file="../data/countries/world_
173       countries.xlsx", row.names = FALSE)
174 # write.xlsx(levels(waste$Country.reporting), file="../data/countries/
175       waste_countries_reporting.xlsx", row.names = FALSE)
176 # write.xlsx(levels(waste$To.or.from.country), file="../data/countries
177       /waste_countries_tofrom.xlsx", row.names = FALSE)
178 # write.xlsx(levels(world_coord$country), file="../data/countries/
179       world_coord_countries.xlsx", row.names = FALSE)
180
181 # Es tradueixen de l'anglès al català els noms dels països amb el
182 # Google Traductor i llegim de nou els fitxers
183 world.countries.trans <- read.csv("../data/countries/world_countries_
184       trans.csv", header = TRUE, sep = ";")
185 levels(world$region) <- world.countries.trans$catalan
186
187 waste.countries.reporting.trans <- read.csv("../data/countries/waste_
188       countries_reporting_trans.csv", header = TRUE, sep = ";")
189 levels(waste$Country.reporting) <- waste.countries.reporting.trans$
190       catalan
191
192 waste.countries.tofrom.trans <- read.csv("../data/countries/waste_
193       countries_tofrom_trans.csv", header = TRUE, sep = ";")
194 levels(waste$To.or.from.country) <- waste.countries.tofrom.trans$
195       catalan

```

```

182 world_coord.trans <- read.csv("../data/countries/world_coord_countries
    _trans.csv", header = TRUE, sep = ";")
183 levels(world_coord$country) <- world_coord.trans$catalan
184
185
186 # Es genera un dataset per les importacions
187 import <- waste %>%
188   filter(Import.export == "Import") %>%
189   rename(From = To.or.from.country,
190          To = Country.reporting,
191          Pop_to = Population,
192          Quantity.in.tonnes_to = Quantity.in.tonnes,
193          Quantity.in.kg.per.capita_to = Quantity.in.kg.per.capita) %>%
194   select(Year, From, To, Pop_to, Quantity.in.tonnes_to, Quantity.in.kg
    .per.capita_to,
195          Disposal.and.recovery.code, General.Basel.Convention.code..Y.
    code) %>%
196   left_join(y = world_coord, by = c("From" = "country")) %>%
197   rename(start_lat = lat,
198          start_long = long) %>%
199   left_join(y = world_coord, by = c("To" = "country")) %>%
200   rename(end_lat = lat,
201          end_long = long) %>%
202   relocate(Year,
203            From, start_lat, start_long,
204            To, end_lat, end_long, Pop_to, Quantity.in.tonnes_to,
205            Quantity.in.kg.per.capita_to,
206            Disposal.and.recovery.code, General.Basel.Convention.code..
    Y.code) %>%
207   left_join(y = basel_conv %>% select(-code), by = c("General.Basel.
    Convention.code..Y.code" = "description"))
208
209 # Es genera un dataset per les exportacions
210 export <- waste %>%
211   filter(Import.export == "Export") %>%
212   rename(From = Country.reporting,
213          To = To.or.from.country,
214          Pop_from = Population,
215          Quantity.in.tonnes_from = Quantity.in.tonnes,
216          Quantity.in.kg.per.capita_from = Quantity.in.kg.per.capita)
    %>%
217   select(Year, From, To, Pop_from, Quantity.in.tonnes_from, Quantity.
    in.kg.per.capita_from,
218          Disposal.and.recovery.code, General.Basel.Convention.code..Y.
    code) %>%
219   left_join(y = world_coord, by = c("From" = "country")) %>%
220   rename(start_lat = lat,
221          start_long = long) %>%
222   left_join(y = world_coord, by = c("To" = "country")) %>%
223   rename(end_lat = lat,
224          end_long = long) %>%
225   relocate(Year,
226            From, start_lat, start_long, Pop_from, Quantity.in.tonnes_
    from, Quantity.in.kg.per.capita_from,
227            To, end_lat, end_long,
228            Disposal.and.recovery.code, General.Basel.Convention.code..
    Y.code) %>%

```

```
228 left_join(y = basel_conv %>% select(-code), by = c("General.Basel.  
      Convention.code..Y.code" = "description"))  
229  
230  
231  
232 # Es mira quin dataset conté més dades agrupant per país i any  
233 a1 <- import %>%  
234   group_by(Year, From, To) %>%  
235   summarise(n = n())  
236  
237 a2 <- export %>%  
238   group_by(Year, From, To) %>%  
239   summarise(n = n())  
240  
241 # Dades que estan en importacions però no en exportacions  
242 anti_join(x = a1, y = a2, by = c("To" = "From", "From" = "To"))  
243  
244 # Dades que estan en exportacions però no en importacions  
245 anti_join(x = a2, y = a1, by = c("From" = "To", "To" = "From"))  
246  
247  
248 # Es guarden tots els objectes creats en un objecte .RData  
249 save.image("preprocessing.RData")
```

Listing B.1: Preprocessament de les dades

C | Taules

País exportador	Quantitat en tones	Quantitat per capita	Població
Alemanya	6047087.96	73.04	82792351
França	5018535.73	74.99	66926166
Suècia	3020722.31	298.48	10120242
Bèlgica	1774153.13	155.65	11398589
Itàlia	1086012.81	17.96	60483973
Noruega	876571.78	165.53	5295619
Àustria	808996.10	91.70	8822267
Espanya	598670.27	12.83	46658447
República Txeca	567611.74	53.50	10610055
Dinamarca	560796.16	97	5781190
Polònia	431814.34	11.37	37976687
Hongria	426379.61	43.60	9778371
Regne Unit	392941.74	5.93	66273576
Portugal	330914.14	32.16	10291027
Eslovàquia	296356.03	54.45	5443120
Bulgària	159291.09	22.59	7050034
Letònia	124388.62	64.30	1934379
Estònia	124038.70	94.03	1319133
Xipre	118488.09	137.10	864236
Grècia	105124.95	9.79	10741165
Finlàndia	94801.87	17.20	5513130
Eslovènia	88927.35	43.03	2066880
Irlanda	33664.20	6.97	4830392
Luxemburg	26485	43.99	602005
Croàcia	19548.72	4.76	4105493
Romania	17621.07	0.90	19530631
Lituània	17552.89	6.25	2808901

Taula C.1: Dades sobre les importacions dels enviaments transfronterers de residus des de tots els països l'any 2018 ordenades per quantat de residus imoprtats en tones.

País	d_v	d_v^{in}	d_v^{out}
Alemanya	2441	2101	340
Bèlgica	1273	918	355
França	1215	704	511
Itàlia	707	141	566
Països Baixos	676	0	676
Regne Unit	556	227	329
Espanya	488	377	111
Suècia	478	304	174
Àustria	476	247	229
Irlanda	458	20	438
Noruega	373	107	266
Luxemburg	343	18	325
Dinamarca	332	182	150
Suïssa	310	0	310
Polònia	201	166	35
Grècia	181	37	144
Eslovènia	162	43	119
Portugal	148	120	28
Finlàndia	123	36	87
Croàcia	104	24	80
Hongria	95	60	35
República Txeca	93	50	43
Bulgària	77	50	27
Lituània	67	23	44
Eslovàquia	63	34	29
Andorra	57	0	57
Estats Units	50	0	50
Romania	50	20	30
Malta	43	0	43
Sèrbia i Montenegro	42	0	42
Estònia	39	28	11
Israel	38	0	38
San Marino	37	0	37
Letònia	36	18	18
Islàndia	20	0	20
Bòsnia i Hercegovina	17	0	17
Xipre	15	5	10
Oman	15	0	15
Brasil	14	0	14
Xina	10	0	10
Emirats Àrabs Units	10	0	10
Mauritània	8	0	8
Mònaco	8	0	8
Canadà	7	0	7
Nigèria	7	0	7
Macedònia	6	0	6
Taiwan	6	0	6
Sud-Àfrica	6	0	6
Malàisia	6	0	6

País	d_v	d_v^{in}	d_v^{out}
Bahrain	6	0	6
Singapur	5	0	5
Algèria	5	0	5
Senegal	5	0	5
Tunísia	5	0	5
Argentina	4	0	4
Nova Zelanda	4	0	4
Rússia	4	0	4
Mèxic	4	0	4
Austràlia	4	0	4
Bielorússia	4	0	4
Turquia	4	0	4
Kuwait	4	0	4
Tailàndia	3	0	3
Japó	3	0	3
Líban	3	0	3
Costa Rica	3	0	3
Aràbia Saudita	3	0	3
Sao Tomé i Príncep	3	0	3
Liechtenstein	2	0	2
Uruguai	2	0	2
Corea del Sud	2	0	2
Jersey	2	0	2
República Democràtica del Congo	2	0	2
Gabon	2	0	2
Jordània	2	0	2
Panamà	2	0	2
Marroc	2	0	2
Costa d'Ivori	2	0	2
Albània	2	0	2
Geòrgia	2	0	2
Moldàvia	2	0	2
Cap Verd	2	0	2
Illes Faroe	2	0	2
Djibouti	2	0	2
Malawi	2	0	2
Barbados	1	0	1
Bahames	1	0	1
Surinam	1	0	1
Sri Lanka	1	0	1
Maurici	1	0	1
Iran	1	0	1
Guinea	1	0	1
Benín	1	0	1
Camerun	1	0	1
Mali	1	0	1
Filipines	1	0	1
Tanzània	1	0	1
Anar	1	0	1
Zàmbia	1	0	1
Ghana	1	0	1
Kazakhstan	1	0	1
Kenya	1	0	1
Bangladesh	1	0	1
Zimbabwe	1	0	1
Swazilàndia	1	0	1
Níger	1	0	1
Qatar	1	0	1
Indonèsia	1	0	1

Taula C.2: Graus de distribució de cada node v (és a dir, els països) del graf G sobre el dataset de les importacions dels enviaments transfronterers de residus des de tots els països l'any 2018. En la primera columna, el grau del cada node; en la segona columna, els graus interns; i, en la tercera columna, els graus externs.

Codi	Descripció	Perillós
Y1	Residus clínics de l'atenció mèdica en hospitals, centres mèdics i clíniques	Si
Y2	Residus de la producció i preparació de productes farmacèutics	Si
Y3	Residus farmacèutics i medicaments	Si
Y4	Residus de la producció, formulació i ús de biocides i fitofarmacèutics	Si
Y5	Residus de la fabricació, formulació i ús de productes químics per a la conservació de la fusta	Si
Y6	Residus de la producció, formulació i ús de dissolvents orgànics	Si
Y7	Residus de les operacions de tractament tèrmic i temperat que contenen cianurs	Si
Y8	Residus d'olis minerals que no son aptes per al seu ús previst originalment	Si
Y9	Residus d'olis/aigua, mescles d'hidrocarburs/aigua, emulsions	Si
Y10	Residus de substàncies i articles que contenen o estan contaminats amb bifenils policlorats (PCB) i/o terfenils policlorats (PCT) i/o bifenils polibromats (PBB)	Si
Y11	Residus residuals residuals derivats de la refinació, la destil·lació i qualsevol tractament pirolític	Si
Y12	Residus de la producció, formulació i ús de tintes, colorants, pigments, pintures, laques, vernissos	Si
Y13	Residus de la producció, formulació i ús de resines, làtex, plastificants, coles/adhesius	Si
Y14	Residus de substàncies químiques derivades d'investigació i desenvolupament o activitats docents que no s'identifiquen i/o son noves i que no es coneixen els efectes sobre l'home i/o el medi ambient	Si
Y15	Residus de caràcter explosiu no subjectes a altres legislacions	Si
Y16	Residus de la producció, formulació i ús de productes químics fotogràfics i materials de processament	Si
Y17	Residus resultants del tractament superficial de metalls i plàstics	Si
Y18	Residus derivats de les operacions d'eliminació de residus industrials	Si
Y19	Carbonis metàl·lics	Si
Y20	Beril·li; compostos de beril·li	Si
Y21	Compostos de crom hexavalents	Si
Y22	Compostos de coure	Si
Y23	Compostos de zinc	Si
Y24	Arsènic; compostos d'arsènic	Si
Y25	Seleni; compostos de seleni	Si
Y26	Cadmí; compostos de cadmi	Si
Y27	Antimoni; compostos d'antimoni	Si
Y28	Tel·luri; compostos de tel·luri	Si
Y29	Mercuri; compostos de mercuri	Si
Y30	Tal·li; compostos de tal·li	Si
Y31	Plom; compostos de plom	Si
Y32	Compostos inorgànics de fluor, excepte el fluorur de calci	Si
Y33	Cianurs inorgànics	Si
Y34	Solucions àcides o àcids en forma sòlida	Si
Y35	Solucions o bases bàsiques en forma sòlida	Si
Y36	Amiant (pols i fibres)	Si
Y37	Compostos orgànics de fòsfor	Si
Y38	Cianurs orgànics	Si
Y39	Fenols; compostos fenòlics inclosos els clorofenols	Si
Y40	Èters	Si
Y41	Solvents orgànics halogenats	Si
Y42	Solvents orgànics, excepte dissolvents halogenats	Si
Y43	Qualsevol congènere de dibenzofurà policlorat	Si
Y44	Qualsevol congènere de dibenz-o-p-dioxina policlorada	Si
Y45	Compostos organohalògens diferents de les substàncies esmentades en aquest annex (per exemple, Y39, Y41, Y42, Y43, Y44)	Si
Y46	Residus recollits de les llars	No
Y47	Residus derivats de la incineració de residus domèstics	No
Not specified	Sense descripció	No
Mix	Mix	No

Taula C.3: Codis, descripcions i criteri de perillositat de la variable Conveni general de Basilea del dataset **waste**.

País	Grau	Proximitat	Interrelació	Vector propi	País	Grau	Proximitat	Interrelació	Vector propi
Alemanya	70	0.00	476.42	0.27	Argentina	2	0.00	0.00	0.03
França	68	0.00	517.03	0.20	Tailàndia	2	0.00	0.00	0.04
Bèlgica	61	0.00	360.02	0.19	Uruguai	2	0.00	0.00	0.03
Itàlia	51	0.00	395.52	0.25	Rússia	2	0.00	0.00	0.03
Regne Unit	45	0.00	390.33	0.26	Corea del Sud	2	0.00	0.00	0.04
Suècia	38	0.00	239.79	0.20	Jersey	2	0.00	0.00	0.04
Polònia	37	0.00	72.53	0.18	Mèxic	2	0.00	0.00	0.03
Espanya	36	0.00	98.89	0.15	Jordània	2	0.00	0.00	0.04
Àustria	35	0.00	104.76	0.19	Panamà	2	0.00	0.00	0.03
Finlàndia	27	0.00	98.69	0.20	Marroc	2	0.00	0.00	0.03
Grècia	24	0.00	70.78	0.19	Mònaco	2	0.00	0.00	0.04
Dinamarca	23	0.00	39.31	0.15	Andorra	2	0.00	0.00	0.03
República Txeca	22	0.00	19.19	0.13	Tunísia	2	0.00	0.00	0.04
Bulgària	22	0.00	10.37	0.10	Costa d'Ivori	2	0.00	0.00	0.04
Noruega	22	0.00	43.07	0.17	Geòrgia	2	0.00	0.00	0.03
Eslovènia	20	0.00	31.07	0.14	Moldàvia	2	0.00	0.00	0.02
Croàcia	19	0.00	8.51	0.18	Barbados	1	0.00	0.00	0.02
Portugal	19	0.00	33.07	0.08	Bahames	1	0.00	0.00	0.02
Romania	19	0.00	73.70	0.09	Surinam	1	0.00	0.00	0.02
Hongria	17	0.00	16.84	0.14	Sri Lanka	1	0.00	0.00	0.02
Lituània	17	0.00	26.03	0.11	Japó	1	0.00	0.00	0.02
Països Baixos	16	0.00	0.00	0.24	Maurici	1	0.00	0.00	0.02
Suïssa	14	0.00	0.00	0.22	Iran	1	0.00	0.00	0.02
Eslovàquia	14	0.00	18.64	0.10	Guinea	1	0.00	0.00	0.02
Irlanda	12	0.00	2.03	0.17	Benín	1	0.00	0.00	0.02
Estònia	11	0.00	4.60	0.07	Camerun	1	0.00	0.00	0.02
Xipre	9	0.00	4.74	0.08	RD del Congo	1	0.00	0.00	0.02
Israel	9	0.00	0.00	0.13	Costa Rica	1	0.00	0.00	0.02
Sèrbia i Montenegro	8	0.00	0.00	0.13	Gabon	1	0.00	0.00	0.02
Letònia	8	0.00	4.10	0.03	Mali	1	0.00	0.00	0.02
Luxemburg	7	0.00	0.00	0.06	Senegal	1	0.00	0.00	0.02
Bòsnia i Hercegovina	6	0.00	0.00	0.10	Filipines	1	0.00	0.00	0.02
Xina	6	0.00	0.00	0.10	Bahrain	1	0.00	0.00	0.02
Islàndia	6	0.00	0.00	0.11	Aràbia Saudita	1	0.00	0.00	0.02
Malta	6	0.00	0.00	0.10	Kuwait	1	0.00	0.00	0.02
Estats Units	5	0.00	0.00	0.08	Albània	1	0.00	0.00	0.02
Singapur	5	0.00	0.00	0.09	San Marino	1	0.00	0.00	0.02
Nigèria	5	0.00	0.00	0.09	Tanzània	1	0.00	0.00	0.02
Emirats Àrabs Units	4	0.00	0.00	0.08	Anar	1	0.00	0.00	0.02
Oman	4	0.00	0.00	0.07	Zàmbia	1	0.00	0.00	0.02
Malàisia	4	0.00	0.00	0.08	Ghana	1	0.00	0.00	0.02
Algèria	4	0.00	0.00	0.07	Kazakhstan	1	0.00	0.00	0.02
Macedònia	3	0.00	0.00	0.05	Kenya	1	0.00	0.00	0.02
Brasil	3	0.00	0.00	0.05	Cap Verd	1	0.00	0.00	0.01
Canadà	3	0.00	0.00	0.05	Bangladesh	1	0.00	0.00	0.01
Taiwan	3	0.00	0.00	0.06	Zimbabwe	1	0.00	0.00	0.01
Nova Zelanda	3	0.00	0.00	0.06	Swazilàndia	1	0.00	0.00	0.01
Sud-Àfrica	3	0.00	0.00	0.05	Illes Faroe	1	0.00	0.00	0.02
Líban	3	0.00	0.00	0.04	Djibouti	1	0.00	0.00	0.02
Mauritània	3	0.00	0.00	0.05	Malawi	1	0.00	0.00	0.02
Austràlia	3	0.00	0.00	0.05	Níger	1	0.00	0.00	0.02
Bielorússia	3	0.00	0.00	0.05	Qatar	1	0.00	0.00	0.02
Turquia	3	0.00	0.00	0.06	Sao Tomé i Príncep	1	0.00	0.00	0.02
Liechtenstein	2	0.00	0.00	0.03	Indonèsia	1	0.00	0.00	0.02

Taula C.4: Centralitat dels nodes de la xarxa d'importacions de tots els països l'any 2018. La taula està ordenada de manera descendent segons el número de graus.

País	k -cores	Color	País	k -cores	Color
Barbados	1	●	Kuwait	4	●
Bahames	1	●	Singapur	5	●
Surinam	1	●	Algèria	5	●
Sri Lanka	1	●	Senegal	5	●
Maurici	1	●	Tunísia	5	●
Iran	1	●	Macedònia	6	●
Guinea	1	●	Taiwan	6	●
Benín	1	●	Sud-Àfrica	6	●
Camerun	1	●	Malàisia	6	●
Mali	1	●	Bahrain	6	●
Filipines	1	●	Canadà	7	●
Tanzània	1	●	Nigèria	7	●
Anar	1	●	Mauritània	8	●
Zàmbia	1	●	Mònaco	8	●
Ghana	1	●	Xina	10	●
Kazakhstan	1	●	Emirats Àrabs Units	10	●
Kenya	1	●	Brasil	14	●
Bangladesh	1	●	Xipre	14	●
Zimbabwe	1	●	Oman	15	●
Swazilàndia	1	●	Bòsnia i Hercegovina	17	●
Níger	1	●	Islàndia	20	●
Qatar	1	●	Letònia	36	●
Indonèsia	1	●	Estònia	36	●
Liechtenstein	2	●	Israel	37	●
Uruguai	2	●	San Marino	37	●
Corea del Sud	2	●	Sèrbia i Montenegro	42	●
Jersey	2	●	Malta	43	●
RD Congo	2	●	Romania	45	●
Gabon	2	●	Lituània	46	●
Jordània	2	●	Bulgària	47	●
Panamà	2	●	Estats Units	49	●
Marroc	2	●	Andorra	57	●
Costa d'Ivori	2	●	Eslovàquia	62	●
Albània	2	●	República Txeca	79	●
Geòrgia	2	●	Hongria	81	●
Moldàvia	2	●	Croàcia	88	●
Cap Verd	2	●	Finlàndia	102	●
Illes Faroe	2	●	Eslovènia	114	●
Djibouti	2	●	Portugal	134	●
Malawi	2	●	Grècia	134	●
Tailàndia	3	●	Polònia	138	●
Japó	3	●	Suïssa	301	●
Líban	3	●	Àustria	301	●
Costa Rica	3	●	Suècia	310	●
Aràbia Saudita	3	●	Dinamarca	310	●
Sao Tomé i Príncep	3	●	Noruega	310	●
Argentina	4	●	Regne Unit	337	●
Nova Zelanda	4	●	Espanya	337	●
Rússia	4	●	Irlanda	337	●
Mèxic	4	●	Luxemburg	342	●
Austràlia	4	●	Itàlia	402	●
Bielorússia	4	●	Alemanya	597	●
Turquia	4	●	França	597	●
			Països Baixos	597	●
			Bèlgica	597	●

Taula C.5: Descomposició k -core per la xarxa d'importacions des de tots els països de l'any 2018. La taula està ordenada de manera ascendent segons el número de nuclis i mostra el mateix color que apareix en el corresponent gràfic.

País	Clústers	d_v	k -cores
Alemanya	1	2441	597
República Txeca	1	93	79
Països Baixos	1	676	597
Polònia	1	201	138
Eslovàquia	1	63	62
Bèlgica	1	1273	597
Brasil	1	14	14
Canadà	1	7	7
Barbados	1	1	1
Àustria	1	476	301
Bahames	1	1	1
Argentina	1	4	4
Tailàndia	1	3	3
Taiwan	1	6	6
Sud-Àfrica	1	6	6
Surinam	1	1	1
Sri Lanka	1	1	1
Corea del Sud	1	2	2
Japó	1	3	3
Jersey	1	2	2
Maurici	1	1	1
Mèxic	1	4	4
Bahrain	1	6	6
Costa d'Ivori	1	2	2
Aràbia Saudita	1	3	3
Nigèria	1	7	7
Kuwait	1	4	4
Itàlia	2	707	402
França	2	1215	597
Suïssa	2	310	301
Estats Units	2	50	49
Emirats Àrabs Units	2	10	10
Uruguai	2	2	2
Portugal	2	148	134
Espanya	2	488	337
Singapur	2	5	5
Luxemburg	2	343	342
Malta	2	43	43
Malàisia	2	6	6
Iran	2	1	1
Guinea	2	1	1
Benín	2	1	1
Camerun	2	1	1
RD del Congo	2	2	2
Costa Rica	2	3	3
Gabon	2	2	2
Jordània	2	2	2
Mali	2	1	1
Senegal	2	5	5
Panamà	2	2	2
Filipines	2	1	1
Marroc	2	2	2
Mònaco	2	8	8
Andorra	2	57	57
Austràlia	2	4	4
Tunísia	2	5	5
San Marino	2	37	37
Tanzània	2	1	1
Anar	2	1	1
Zàmbia	2	1	1
Ghana	2	1	1

País	Clústers	d_v	k -cores
Kazakhstan	2	1	1
Kenya	2	1	1
Cap Verd	2	2	2
Swazilàndia	2	1	1
Finlàndia	3	123	102
Regne Unit	3	556	337
Suècia	3	478	310
Dinamarca	3	332	310
Xipre	3	15	14
Nova Zelanda	3	4	4
Noruega	3	373	310
Rússia	3	4	4
Irlanda	3	458	337
Islàndia	3	20	20
Líban	3	3	3
Lituània	3	67	46
Letònia	3	36	36
Estònia	3	39	36
Mauritània	3	8	8
Bielorússia	3	4	4
Geòrgia	3	2	2
Illes Faroe	3	2	2
Djibouti	3	2	2
Malawi	3	2	2
Níger	3	1	1
Qatar	3	1	1
Sao Tomé i Príncep	3	3	3
Indonèsia	3	1	1
Croàcia	4	104	88
Hongria	4	95	81
Eslovènia	4	162	114
Sèrbia i Montenegro	4	42	42
Liechtenstein	4	2	2
Macedònia	4	6	6
Bòsnia i Hercegovina	4	17	17
Xina	4	10	10
Bulgària	4	77	47
Oman	4	15	15
Israel	4	38	37
Grècia	4	181	134
Romania	4	50	45
Algèria	4	5	5
Turquia	4	4	4
Albània	4	2	2
Moldàvia	4	2	2
Bangladesh	4	1	1
Zimbabwe	4	1	1

Taula C.6: Clúster, grau dels nodes i nucli dels nodes per a les regions de la xarxa d'importacions des de tots els països de l'any 2018

D | Anàlisis descriptives

```
1 # Paquets necessaris
2 library(xtable)
3 library(network)
4 library(sna)
5 library(tidyverse)
6 library(ape)
7 library(scales)
8
9 # Es carreguen les dades des d'altres scripts
10 source("visNetwork.R")
11
12
13 #####
14 # Anàlisi descriptiva #
15 #####
16
17 # Es crea un objecte de classe igraph
18 dades <- network(2018, "Tots")
19 g.import <- graph_from_data_frame(d = dades$vis.links, vertices =
    dades$vis.nodes, directed = TRUE)
20
21 class(g.import)
22 summary(g.import)
23
24 # Es pot accedir als seus atributs
25 E(g.import)$weight
26 E(g.import)$hazardous
27 V(g.import)$Pop_to
28
29 # S'assigna un nom
30 g.import$name <- "Importacions"
31
32 # El número de nodes
33 vcount(g.import)
34
35 # El número d'enllaços
36 ecoun(g.import)
37
38 # És un multi-graf dirigit, ponderat i està connectat
39 is_simple(g.import)
40 is_directed(g.import)
41 is_weighted(g.import)
42 is_connected(g.import)
43
44 # El graf està feblement connectat perquè el seu graf subjacent està
```

```

connectat
45 is_connected(g.import, mode = "weak")
46 is_connected(g.import, mode = "strong")
47
48 # Nodes adjacents (veïns) a tots els països
49 neighbors(g.import, 1)
50 unique(neighbors(g.import, 1))
51
52 # Visualització de grafs
53 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "igraph", "
    importacions", "tots", "2018"), width = 1000, height = 1000, res =
    100)
54 par(mfrow = c(2,2))
55 igraph_options(vertex.size = 5, vertex.label = NA, edge.arrow.size =
    0.05, cex.main = 3)
56 plot(g.import, layout = layout_in_circle, main = "Circular")
57 plot(g.import, layout = layout_with_fr, main = "Fruchterman i Reingol"
    )
58 plot(g.import, layout = layout_as_tree(g.import, circular = TRUE),
    main = "Arbre (disp. circular)")
59 plot(g.import, layout = layout_as_tree, main = "Arbre")
60 dev.off()
61
62 # Grau dels nodes
63 igraph::degree(g.import)
64 igraph::degree(g.import, mode = "in")
65 igraph::degree(g.import, mode = "out")
66
67 # Histograma del grau dels nodes
68 degree.hist <- ggplot(data.frame(degree = igraph::degree(g.import)),
    aes(x = degree)) +
69   geom_histogram(binwidth = 100, fill = "#d5feff", color = "#00CED1",
    size = 0.3) +
70   scale_x_continuous(breaks = seq(0, max(igraph::degree(g.import)),
    250)) +
71   labs(x = "Grau dels vèrtexs", y = "Freqüència") +
72   theme_bw()
73
74 ggsave(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "degree", "
    importacions", "tots", "2018"),
    plot = degree.hist, width = 15, height = 10, units = "cm")
75
76
77
78 # Taula del grau dels nodes
79 degree.table <- data.frame(degree = igraph::degree(g.import),
    degree.in = igraph::degree(g.import, mode =
    "in"),
80   degree.out = igraph::degree(g.import, mode
    = "out")) %>%
81   arrange(-degree) %>%
82   rownames_to_column()
83
84
85 bold <- function(x) {paste('\\textbf{' , x, '}', sep = '')}
86 print(xtable(degree.table %>% rename("$d_v$" = "degree",
    "$d^{in}_v$" = "degree.in",
    "$d^{out}_v$" = "degree.out"),
87   digits = 0),
88

```

```

90     sanitize.colnames.function = bold,
91     booktabs = TRUE,
92     floating = TRUE,
93     file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "degree"
94         , "importacions", "tots", "2018"))
95
96 # Força dels nodes
97 strength.hist <- ggplot(data.frame(strength = strength(g.import)), aes
98     (x = strength)) +
99     geom_histogram(binwidth = 100000, fill = "#f1ffd5", color = "#c0ea6c"
100         , size = 0.3) +
101     scale_x_continuous(breaks = seq(0, max(strength(g.import)), 1000000)
102         ) +
103     labs(x = "Força dels vèrtexs", y = "Freqüència") +
104     theme_bw()
105
106 ggsave(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "strength", "
107     importacions", "tots", "2018"),
108     plot = strength.hist, width = 15, height = 10, units = "cm")
109
110 # Es simplifica el graf per a poder calcular la mitjana dels graus
111 # dels nodes veïns
112 g.import.simple <- igraph::simplify(g.import, remove.multiple = TRUE,
113     remove.loops = TRUE)
114 is_simple(g.import.simple)
115
116 # Taula del grau dels nodes
117 degree.table.hazard <- data.frame(degree = igraph::degree(g.import.
118     hazard),
119
120         degree.in = igraph::degree(g.import.hazard,
121             mode = "in"),
122         degree.out = igraph::degree(g.import.hazard
123             , mode = "out")) %>%
124
125     arrange(-degree) %>%
126     rownames_to_column()
127
128 degree.table.nonhazard <- data.frame(degree = igraph::degree(g.import.
129     nonhazard),
130
131         degree.in = igraph::degree(g.import.
132             nonhazard, mode = "in"),
133         degree.out = igraph::degree(g.import
134             .nonhazard, mode = "out")) %>%
135
136     arrange(-degree) %>%
137     rownames_to_column()
138
139 # Mitjana dels graus dels nodes veïns
140 a.nn.deg <- knn(g.import.simple, V(g.import.simple))$knn
141 a.nn.deg.data <- data.frame(knn = a.nn.deg, degree = degree(g.import.
142     simple))
143
144 knn.plot <- ggplot(a.nn.deg.data, aes(x = degree, y = knn)) +
145     geom_point() +
146     scale_x_continuous(trans = "log10") +
147     scale_y_continuous(trans = "log10") +
148     geom_smooth(method = lm, se = FALSE, color = "red3") +
149     labs(x = "Grau dels vèrtexs", y = "Grau en mitjana del veí") +
150     theme_classic()

```

```

133
134 ggsave(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "knn", "
      importacions", "tots", "2018"),
135         plot = knn.plot, width = 15, height = 10, units = "cm")
136
137 # Centralitat dels nodes
138 A <- as_adjacency_matrix(g.import, sparse = FALSE)
139 g <- network::as.network.matrix(A)
140
141 write.csv(A, file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.csv", "
      adjacencia", "importacions", "tots", "2018"))
142
143 print(xtable(A, digits = 2),
144       sanitize.colnames.function = bold,
145       include.rownames = FALSE,
146       booktabs = TRUE,
147       floating = TRUE,
148       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
      adjacencia", "importacions", "tots", "2018"))
149
150 special <- which(names(igraph::degree(g.import)) == c("Alemanya", "
      Bèlgica", "França"))
151 colors <- rep("blue", length(igraph::degree(g.import)))
152 colors[special] <- "yellow"
153
154 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "centralitats", "
      importacions", "tots", "2018"), width = 1000, height = 1000, res =
      75)
155 par(mfrow = c(2,2))
156 sna::gplot.target(g, sna::degree(g),
157                   main = "Graus",
158                   cex.main = 2,
159                   circ.col = "skyblue",
160                   circ.lab = FALSE,
161                   usearrows = FALSE,
162                   vertex.col = colors,
163                   edge.col = "darkgray")
164
165 sna::gplot.target(g, sna::betweenness(g),
166                   main = "Interrelació",
167                   cex.main = 2,
168                   circ.col = "skyblue",
169                   circ.lab = FALSE,
170                   usearrows = FALSE,
171                   vertex.col = colors,
172                   edge.col = "darkgray")
173
174 sna::gplot.target(g, sna::closeness(g),
175                   main = "Proximitat",
176                   cex.main = 2,
177                   circ.col = "skyblue",
178                   circ.lab = FALSE,
179                   usearrows = FALSE,
180                   vertex.col = colors,
181                   edge.col = "darkgray")
182
183 sna::gplot.target(g, sna::evcent(g),

```

```

184         main = "Vector propi",
185         cex.main = 2,
186         circ.col = "skyblue",
187         circ.lab = FALSE,
188         usearrows = FALSE,
189         vertex.col = colors,
190         edge.col = "darkgray")
191 dev.off()
192
193
194 centralitats <- data.frame(nom = V(g.import)$name,
195                             grau = sna::degree(g),
196                             proximitat = sna::closeness(g),
197                             interrelacio = sna::betweenness(g),
198                             vep = sna::evcent(g)) %>%
199     arrange(desc(grau))
200
201 print(xtable(centralitats %>% rename("País" = "nom",
202                                     "Grau" = "grau",
203                                     "Proximitat" = "proximitat",
204                                     "Interrelació" = "interrelacio",
205                                     "Vector propi" = "vep"),
206         digits = 2),
207       sanitize.colnames.function = bold,
208       include.rownames = FALSE,
209       booktabs = TRUE,
210       floating = TRUE,
211       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
212                       centralitats", "importacions", "tots", "2018"))
213
214 # Caracterització dels enllaços
215 eb <- edge_betweenness(g.import)
216 E(g.import)[order(eb, decreasing=T)[1:3]]
217
218 # Cliques
219 table(sapply(cliques(g.import), length))
220 cliques(g.import)[sapply(cliques(g.import), length) == 11]
221 table(sapply(max_cliques(g.import), length))
222 clique_num(g.import)
223
224 # Cores
225 cores <- coreness(g.import)
226
227 # S'assigna un color a cada número de cores
228 cores.d <- as.data.frame(cores) %>% arrange(cores)
229 valor.previ <- 0
230 n <- length(unique(cores.d$cores))
231 color <- colorRampPalette(brewer.pal(8, "Spectral"))(n)
232 color.tikz <- sprintf("\\color[HTML]{%s}\\bullet", str_sub(color,
233     start = -6))
234
235 color.actual <- 0
236
237 for(i in 1:nrow(cores.d)){
238     if(cores.d$cores[i] != valor.previ){
239         color.actual <- color.actual + 1
240     }
241 }

```

```

239 cores.d$color[i] <- color[color.actual]
240 cores.d$color.tikz[i] <- color.tikz[color.actual]
241 valor.previ <- cores.d$cores[i]
242 }
243
244 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "coreness", "
    importacions", "tots", "2018"), width = 1000, height = 1000, res =
    75)
245 sna::gplot.target(g, cores.d$cores,
246                   circ.col = "skyblue",
247                   circ.lwd = 3,
248                   circ.lab = FALSE,
249                   usearrows = FALSE,
250                   vertex.col = cores.d$color,
251                   edge.col = "darkgray")
252
253 legend("right", legend = unique(cores.d$cores), pch = 19, col = unique
    (cores.d$color),
254       inset = 0, bty = "n", cex = 1.5, title = "k-cores")
255 dev.off()
256
257 print(xtable(cores.d %>% rownames_to_column() %>% select(rowname,
    cores, color.tikz) %>% rename("País" = "rowname", "$k$-cores" = "
    cores", "Color" = "color.tikz"),
258       digits = 0),
259       sanitize.colnames.function = bold,
260       booktabs = TRUE,
261       floating = TRUE,
262       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
    coreness", "importacions", "tots", "2018"))
263
264 # Díades
265 dyad_census(g.import.simple)
266
267 # Densitat global de la xarxa
268 ecount(g.import)
269 vcount(g.import)
270
271 density <- function(g){
272   num <- abs(ecount(g))
273   if(is_directed(g)){
274     den <- abs(vcount(g))*(abs(vcount(g))-1)
275   }
276   else{
277     den <- abs(vcount(g))*(abs(vcount(g))-1)/2
278   }
279   return(num/den)
280 }
281
282 density(g.import)
283 density(g.import.simple)
284
285 # És el mateix que:
286 edge_density(g.import)
287 edge_density(g.import.simple)
288
289 # Densitat de subgrafs

```

```

290 c(which(V(g.import)$name == c("Alemanya", "Bèlgica", "França")), 13) #
    el 13 fa referència a Països Baixos
291
292 ego_alemanya <- induced_subgraph(g.import.simple, vids = neighborhood(
    g.import.simple, 1, 1)[[1]])
293 ego_belgica <- induced_subgraph(g.import.simple, vids = neighborhood(g
    .import.simple, 6, 6)[[1]])
294 ego_franca <- induced_subgraph(g.import.simple, vids = neighborhood(g.
    import.simple, 20, 20)[[1]])
295 ego_pb <- induced_subgraph(g.import.simple, vids = neighborhood(g.
    import.simple, 13, 13)[[1]])
296 edge_density(induced_subgraph(g.import.simple, vids = neighborhood(g.
    import.simple, 1, 13)[[1]]))
297
298 edge_density(ego_alemanya)
299 edge_density(ego_belgica)
300 edge_density(ego_franca)
301 edge_density(ego_pb)
302
303 # Transitivitat/coeficient d'agrupació (clustering)
304 transitivity(g.import)
305 length(triangles(g.import))
306
307 # Reciprocitat
308 reciprocity(g.import.simple, mode = "default")
309 reciprocity(g.import.simple, mode = "ratio")
310
311 # Connecitivitat
312 comps <- decompose(g.import)
313 table(sapply(comps, vcount))
314 (108/vcount(g.import))*100
315
316 # Distància mitjana
317 mean_distance(g.import)
318 log(vcount(g.import))
319
320 # Càlcul del diàmetre: el valor de la distància més gran
321 diameter(g.import, weights = NA)
322
323 # Transitivitat
324 transitivity(g.import)
325
326 # Clustering
327 g.import.un <- graph_from_data_frame(d = dades$vis.links, vertices =
    dades$vis.nodes, directed = FALSE)
328 g.import.un.simple <- simplify(g.import.un, remove.multiple = TRUE,
    remove.loops = TRUE)
329
330 kc <- cluster_fast_greedy(g.import.un.simple)
331 length(kc)
332 sizes(kc)
333 membership(kc)
334
335 memb.table <- data.frame(membership = as.matrix(membership(kc))) %>%
336   arrange(membership) %>%
337   rownames_to_column()
338

```

```

339 data <- left_join(x = memb.table, y = degree.table %>% select(-degree.
      in, -degree.out)) %>%
340 left_join(y = cores.d %>% rownames_to_column() %>% select(rowname,
      cores))
341
342 data.table <- data %>%
343 group_by(membership) %>%
344 summarise(sum.degrees = sum(degree),
345           sum.cores = sum(cores))
346
347 print(xtable(data %>% rename("País" = "rowname",
      "Clúster" = "membership",
348     "$d_v$" = "degree",
349     "Nuclis" = "cores"),
350       digits = 0),
351       sanitize.colnames.function = bold,
352       include.rownames = FALSE,
353       booktabs = TRUE,
354       floating = TRUE,
355       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
356         membership", "importacions", "tots", "2018"))
357
358 print(xtable(data.table %>% rename("Clúster" = "membership",
      "$\\sum d_v$" = "sum.degrees",
359     "$\\sum k$" = "sum.cores"),
360       digits = 0),
361       sanitize.colnames.function = bold,
362       include.rownames = FALSE,
363       booktabs = TRUE,
364       floating = TRUE,
365       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
366         membership.summary", "importacions", "tots", "2018"))
367
368 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "clustering", "
      importacions", "tots", "2018"), width = 2000, height = 2000, res =
      100)
369 plot(kc, g.import.un.simple)
370 dev.off()
371
372 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "dendrograma", "
      importacions", "tots", "2018"), width = 2000, height = 2000, res =
      100)
373 plot_dendrogram(kc)
374 dev.off()
375
376 # Assortativitat
377 assortativity_degree(g.import.simple)
378
379
380
381
382
383
384 #####
385 # Anàlisi descriptiva II #
386 #####
387

```

```

388 # Conveni general de Basilea: taula
389 print(xtable(basel_conv %>%
390             mutate_at(.vars = vars(hazardous),
391                       recode, "1" = "Sí", "0" = "No") %>%
392             rename("Codi" = "code",
393                   "Descripció" = "description",
394                   "Perillós" = "hazardous"),
395             digits = 0),
396       sanitize.colnames.function = bold,
397       include.rownames = FALSE,
398       booktabs = TRUE,
399       floating = TRUE,
400       file = sprintf("../LaTeX/tables/taula-%s.tex", "conveni_basilea"
401                       ))
402 # Conveni general de Basilea: plot per a la variable en general
403 basel_conv.data <- basel_conv %>%
404   group_by(hazardous) %>%
405   mutate(cnt = n(),
406          pct = percent(cnt / nrow(.), accuracy = 1)) %>%
407   select(cnt, pct) %>%
408   unique() %>%
409   ungroup() %>%
410   mutate(hazardous = factor(hazardous, levels = c(0, 1), labels = c("
411                               No", "Sí")))
412 basel_conv.plot <- ggplot(basel_conv.data, aes(x = "", y = cnt, fill =
413           factor(hazardous))) +
414   geom_bar(width = 1, stat = "identity") +
415   coord_polar("y", start = 0) +
416   scale_fill_manual(values = c("#00CED1", "#057dA4")) +
417   geom_text(aes(label = sprintf("%d \n %s", cnt, pct)),
418             position = position_stack(vjust = 0.5),
419             col="white") +
420   theme_void() +
421   labs(fill="Perillositat dels residus")
422 ggsave(sprintf("../LaTeX/figures/plot-%s.png", "conveni_basilea"),
423          plot = basel_conv.plot, width = 15, height = 10, units = "cm")
424
425 # Conveni general de Basilea: plot per a les importacions
426 basel_conv.import <- import %>%
427   filter(Year == 2018) %>%
428   group_by(hazardous) %>%
429   mutate(cnt = n(),
430          pct = percent(cnt / nrow(.), accuracy = 1)) %>%
431   select(cnt, pct) %>%
432   unique() %>%
433   ungroup() %>%
434   mutate(hazardous = factor(hazardous, levels = c(0, 1), labels = c("
435                               No", "Sí")))
436 basel_conv.import.plot <- ggplot(basel_conv.import, aes(x = "", y =
437           cnt, fill = factor(hazardous))) +
438   geom_bar(width = 1, stat = "identity") +
439   coord_polar("y", start = 0) +
440   scale_fill_manual(values = c("#00CED1", "#057dA4")) +

```

```

440 geom_text(aes(label = sprintf("%d \n %s", cnt, pct)),
441           position = position_stack(vjust = 0.5),
442           col="white") +
443 theme_void() +
444 labs(fill="Perillositat dels residus")
445
446 ggsave(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "conveni_
447             basilea", "importacions", "tots", "2018"),
448           plot = basel_conv.import.plot, width = 15, height = 10, units =
449             "cm")
450
451 # Es crea una funció network nova per a filtrar els datasets segons la
452   perillositat dels residus
453 network2 <- function(any, pais, perillositat){
454   if(prod(dim(import %>% filter(if(any(pais == "Tots")) Year == any
455     else Year == any & From %in% pais))) == 0)
456     return(NULL)
457
458   else{
459     links <- import %>%
460       filter(if(any(pais == "Tots")) Year == any & hazardous ==
461         perillositat
462         else Year == any & From %in% pais & hazardous ==
463           perillositat) %>%
464       select(From, To, Quantity.in.tonnes_to, Quantity.in.kg.per.
465         capita_to, hazardous) %>%
466       rename(weight = Quantity.in.tonnes_to, from = From, to = To)
467
468     nodes <- data.frame(country = union(links$from, links$to)) %>%
469       left_join(y = import %>% filter(Year == any) %>% select(To, Pop_
470         to) %>% distinct(),
471         by = c("country" = "To")) %>%
472       rename(label = country)
473
474     vis.nodes <- nodes
475     vis.links <- links
476
477     #####
478     # Nodes #
479     #####
480
481     vis.nodes$id <- vis.nodes$label
482     vis.nodes$shadow <- FALSE
483     vis.nodes$borderWidth <- 2
484     vis.nodes$color.background <- ifelse(nodes$label %in% pais, "red",
485       "blue")
486     vis.nodes$color.border <- "black"
487     vis.nodes$group <- ifelse(nodes$label %in% pais, "Pa\u00EFs
488       importadors", "Pa\u00EFs exportadors")
489     vis.nodes$margin <- 1
490
491     #####
492     # Links #
493     #####

```

```

487
488   vis.links$id <- 1:nrow(vis.links)
489   vis.links$width <- links$weight/10000
490   vis.links$color <- ifelse(links$hazardous == 1, "#802621", "#1
      b852e")
491   vis.links$arrows <- "from"
492   vis.links$smooth <- FALSE
493   vis.links$shadow <- FALSE
494
495
496   return(list("vis.links" = vis.links, "vis.nodes" = vis.nodes))
497
498 }
499 }
500
501 # Es creen 2 objectes de classe igraph
502 dades.hazard <- network2(2018, "Tots", 1)
503 dades.nonhazard <- network2(2018, "Tots", 0)
504
505 g.import.hazard <- graph_from_data_frame(d = dades.hazard$vis.links,
      vertices = dades.hazard$vis.nodes, directed = TRUE)
506 g.import.nonhazard <- graph_from_data_frame(d = dades.nonhazard$vis.
      links, vertices = dades.nonhazard$vis.nodes, directed = TRUE)
507
508 # El número de nodes
509 vcount(g.import.hazard)
510 vcount(g.import.nonhazard)
511
512 # El número d'enllaços
513 ecount(g.import.hazard)
514 ecount(g.import.nonhazard)
515
516 # Visualització de grafs
517 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "igraph", "
      importacions", "hazard", "2018"), width = 1000, height = 1000, res
      = 100)
518 par(mfrow = c(2,2))
519 igraph_options(vertex.size = 5, vertex.label = NA, edge.arrow.size =
      0.05, cex.main = 3)
520 plot(g.import.hazard, layout = layout_in_circle, main = "Circular")
521 plot(g.import.hazard, layout = layout_with_fr, main = "Fruchterman i
      Reingol")
522 plot(g.import.hazard, layout = layout_as_tree(g.import.hazard,
      circular = TRUE), main = "Arbre (disp. circular)")
523 plot(g.import.hazard, layout = layout_as_tree, main = "Arbre")
524 dev.off()
525
526 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "igraph", "
      importacions", "nonhazard", "2018"), width = 1000, height = 1000,
      res = 100)
527 par(mfrow = c(2,2))
528 igraph_options(vertex.size = 5, vertex.label = NA, edge.arrow.size =
      0.05, cex.main = 3)
529 plot(g.import.nonhazard, layout = layout_in_circle, main = "Circular")
530 plot(g.import.nonhazard, layout = layout_with_fr, main = "Fruchterman
      i Reingol")
531 plot(g.import.nonhazard, layout = layout_as_tree(g.import.nonhazard,

```

```

        circular = TRUE), main = "Arbre (disp. circular)")
532 plot(g.import.nonhazard, layout = layout_as_tree, main = "Arbre")
533 dev.off()
534
535 # Grau dels nodes
536 sort(igraph::degree(g.import.hazard), decreasing = TRUE)[1:5]
537 sort(igraph::degree(g.import.hazard, mode = "in"), decreasing = TRUE)
    [1:5]
538 sort(igraph::degree(g.import.hazard, mode = "out"), decreasing = TRUE)
    [1:5]
539
540 sort(igraph::degree(g.import.nonhazard), decreasing = TRUE)[1:5]
541 sort(igraph::degree(g.import.nonhazard, mode = "in"), decreasing =
    TRUE)[1:5]
542 sort(igraph::degree(g.import.nonhazard, mode = "out"), decreasing =
    TRUE)[1:5]
543
544 # Nodes adjacents (veïns) a tots els països
545 neighbors(g.import.hazard, 1)
546 unique(neighbors(g.import.hazard, 1))
547
548 neighbors(g.import.nonhazard, 1)
549 unique(neighbors(g.import.nonhazard, 1))
550
551 # Es simplifiquen els grafs per a poder calcular la mitjana dels graus
    dels nodes veïns
552 g.import.hazard.simple <- igraph::simplify(g.import.hazard, remove.
    multiple = TRUE, remove.loops = TRUE)
553 g.import.nonhazard.simple <- igraph::simplify(g.import.nonhazard,
    remove.multiple = TRUE, remove.loops = TRUE)
554
555 is_simple(g.import.hazard.simple)
556 is_simple(g.import.nonhazard.simple)
557
558 # Mitjana dels graus dels nodes veïns
559 a.nn.deg <- knn(g.import.hazard.simple, V(g.import.hazard.simple))$knn
560 a.nn.deg.data <- data.frame(knn = a.nn.deg, degree = igraph::degree(g.
    import.hazard.simple))
561
562 knn.plot <- ggplot(a.nn.deg.data, aes(x = degree, y = knn)) +
563   geom_point() +
564   scale_x_continuous(trans = "log10") +
565   scale_y_continuous(trans = "log10") +
566   geom_smooth(method = lm, se = FALSE, color = "red3") +
567   labs(x = "Grau dels vèrtexs", y = "Grau en mitjana del veí") +
568   theme_classic()
569
570 ggsave(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "knn", "
    importacions", "hazard", "2018"),
    plot = knn.plot, width = 15, height = 10, units = "cm")
571
572 ##
573
574
575 a.nn.deg <- knn(g.import.nonhazard.simple, V(g.import.nonhazard.simple
    ))$knn
576 a.nn.deg.data <- data.frame(knn = a.nn.deg, degree = igraph::degree(g.
    import.nonhazard.simple))

```

```

577
578 knn.plot <- ggplot(a.nn.deg.data, aes(x = degree, y = knn)) +
579   geom_point() +
580   scale_x_continuous(trans = "log10") +
581   scale_y_continuous(trans = "log10") +
582   geom_smooth(method = lm, se = FALSE, color = "red3") +
583   labs(x = "Grau dels vèrtexs", y = "Grau en mitjana del veí") +
584   theme_classic()
585
586 ggsave(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "knn", "
    importacions", "nonhazard", "2018"),
587   plot = knn.plot, width = 15, height = 10, units = "cm")
588
589
590 # Centralitat dels nodes
591 A <- as_adjacency_matrix(g.import.hazard, sparse = FALSE)
592 g.hazard <- network::as.network.matrix(A)
593
594 write.csv(A, file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.csv", "
    adjacencia", "importacions", "hazard", "2018"))
595
596 special <- which(names(igraph::degree(g.import.hazard)) == c("Alemanya
    ", "Bèlgica", "França"))
597 colors <- rep("blue", length(igraph::degree(g.import.hazard)))
598 colors[special] <- "yellow"
599
600 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "centralitats", "
    importacions", "hazard", "2018"), width = 1000, height = 1000, res
    = 75)
601 par(mfrow = c(2,2))
602 sna::gplot.target(g.hazard, sna::degree(g.hazard),
603   main = "Graus",
604   cex.main = 2,
605   circ.col = "skyblue",
606   circ.lab = FALSE,
607   usearrows = FALSE,
608   vertex.col = colors,
609   edge.col = "darkgray")
610
611 sna::gplot.target(g.hazard, sna::betweenness(g.hazard),
612   main = "Interrelació",
613   cex.main = 2,
614   circ.col = "skyblue",
615   circ.lab = FALSE,
616   usearrows = FALSE,
617   vertex.col = colors,
618   edge.col = "darkgray")
619
620 sna::gplot.target(g.hazard, sna::closeness(g.hazard),
621   main = "Proximitat",
622   cex.main = 2,
623   circ.col = "skyblue",
624   circ.lab = FALSE,
625   usearrows = FALSE,
626   vertex.col = colors,
627   edge.col = "darkgray")
628

```

```

629 sna::gplot.target(g.hazard, sna::evcent(g.hazard),
630                   main = "Vector propi",
631                   cex.main = 2,
632                   circ.col = "skyblue",
633                   circ.lab = FALSE,
634                   usearrows = FALSE,
635                   vertex.col = colors,
636                   edge.col = "darkgray")
637 dev.off()
638
639 ##
640
641 A <- as_adjacency_matrix(g.import.nonhazard, sparse = FALSE)
642 g.nonhazard <- network::as.network.matrix(A)
643
644 write.csv(A, file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.csv", "
645   adjacencia", "importacions", "nonhazard", "2018"))
646
647 special <- which(names(igraph::degree(g.import.nonhazard)) == c("
648   Alemanya", "Bèlgica", "França"))
649 colors <- rep("blue", length(igraph::degree(g.import.nonhazard)))
650 colors[special] <- "yellow"
651
652 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "centralitats", "
653   importacions", "nonhazard", "2018"), width = 1000, height = 1000,
654       res = 75)
655 par(mfrow = c(2,2))
656 sna::gplot.target(g.nonhazard, sna::degree(g.nonhazard),
657                   main = "Graus",
658                   cex.main = 2,
659                   circ.col = "skyblue",
660                   circ.lab = FALSE,
661                   usearrows = FALSE,
662                   vertex.col = colors,
663                   edge.col = "darkgray")
664
665 sna::gplot.target(g.nonhazard, sna::betweenness(g.nonhazard),
666                   main = "Interrelació",
667                   cex.main = 2,
668                   circ.col = "skyblue",
669                   circ.lab = FALSE,
670                   usearrows = FALSE,
671                   vertex.col = colors,
672                   edge.col = "darkgray")
673
674 sna::gplot.target(g.nonhazard, sna::closeness(g.nonhazard),
675                   main = "Proximitat",
676                   cex.main = 2,
677                   circ.col = "skyblue",
678                   circ.lab = FALSE,
679                   usearrows = FALSE,
680                   vertex.col = colors,
681                   edge.col = "darkgray")
682
683 sna::gplot.target(g.nonhazard, sna::evcent(g.nonhazard),
684                   main = "Vector propi",
685                   cex.main = 2,

```



```

682         circ.col = "skyblue",
683         circ.lab = FALSE,
684         usearrows = FALSE,
685         vertex.col = colors,
686         edge.col = "darkgray")
687 dev.off()
688
689 # Caracterització dels enllaços
690 eb <- edge_betweenness(g.import.hazard)
691 E(g.import.hazard)[order(eb, decreasing=T)[1:3]]
692
693 eb <- edge_betweenness(g.import.nonhazard)
694 E(g.import.nonhazard)[order(eb, decreasing=T)[1:3]]
695
696 # Cliques
697 table(sapply(cliques(g.import.hazard), length))
698 cliques(g.import.hazard)[sapply(cliques(g.import.hazard), length) ==
699   9]
700 clique_num(g.import.hazard)
701
702 table(sapply(cliques(g.import.nonhazard), length))
703 cliques(g.import.nonhazard)[sapply(cliques(g.import.nonhazard), length
704   ) == 9]
704 table(sapply(max_cliques(g.import.nonhazard), length))
705 clique_num(g.import.nonhazard)
706
707 # Cores
708 cores.hazard <- coreness(g.import.hazard)
709
710 # S'assigna un color a cada número de cores
711 cores.d.hazard <- as.data.frame(cores.hazard) %>% arrange(cores.hazard
712   )
713 valor.previ <- 0
714 n <- length(unique(cores.d.hazard$cores))
715 color <- colorRampPalette(brewer.pal(8, "Spectral"))(n)
716 color.tikz <- sprintf("\\color[HTML]{%s}\\bullet", str_sub(color,
717   start = -6))
718 color.actual <- 0
719
720 for(i in 1:nrow(cores.d.hazard)){
721   if(cores.d.hazard$cores[i] != valor.previ){
722     color.actual <- color.actual + 1
723   }
724   cores.d.hazard$color[i] <- color[color.actual]
725   cores.d.hazard$color.tikz[i] <- color.tikz[color.actual]
726   valor.previ <- cores.d.hazard$cores[i]
727 }
728
729 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "coreness", "
730   importacions", "hazard", "2018"), width = 1000, height = 1000, res
731   = 75)
732 sna::gplot.target(g.hazard, cores.d.hazard$cores,
733   circ.col = "skyblue",
734   circ.lwd = 3,
735   circ.lab = FALSE,
736   usearrows = FALSE,

```

```

733         vertex.col = cores.d.hazard$color,
734         edge.col = "darkgray")
735
736 legend("right", legend = unique(cores.d.hazard$cores), pch = 19, col =
       unique(cores.d.hazard$color),
737       inset = 0, bty = "n", cex = 1.5, title = "k-cores")
738 dev.off()
739
740 print(xtable(cores.d.hazard %>% rownames_to_column() %>% select(
       rowname, cores.hazard, color.tikz) %>% rename("País" = "rowname",
       "$k$-cores" = "cores.hazard", "Color" = "color.tikz"),
741       digits = 0),
742       sanitize.colnames.function = bold,
743       booktabs = TRUE,
744       floating = TRUE,
745       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
       coreness", "importacions", "hazard", "2018"))
746
747 ##
748
749 cores.nonhazard <- coreness(g.import.nonhazard)
750
751 # S'assigna un color a cada número de cores
752 cores.d.nonhazard <- as.data.frame(cores.nonhazard) %>% arrange(cores.
       nonhazard)
753 valor.previ <- 0
754 n <- length(unique(cores.d.nonhazard$cores))
755 color <- colorRampPalette(brewer.pal(8, "Spectral"))(n)
756 color.tikz <- sprintf("\\color[HTML]{%s}\\bullet", str_sub(color,
       start = -6))
757 color.actual <- 0
758
759 for(i in 1:nrow(cores.d.nonhazard)){
760   if(cores.d.nonhazard$cores[i] != valor.previ){
761     color.actual <- color.actual + 1
762   }
763   cores.d.nonhazard$color[i] <- color[color.actual]
764   cores.d.nonhazard$color.tikz[i] <- color.tikz[color.actual]
765   valor.previ <- cores.d.nonhazard$cores[i]
766 }
767
768 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "coreness", "
       importacions", "nonhazard", "2018"), width = 1000, height = 1000,
       res = 75)
769 sna::gplot.target(g.nonhazard, cores.d.nonhazard$cores,
770                   circ.col = "skyblue",
771                   circ.lwd = 3,
772                   circ.lab = FALSE,
773                   usearrows = FALSE,
774                   vertex.col = cores.d.nonhazard$color,
775                   edge.col = "darkgray")
776
777 legend("right", legend = unique(cores.d.nonhazard$cores), pch = 19,
       col = unique(cores.d.nonhazard$color),
778       inset = 0, bty = "n", cex = 1.5, title = "k-cores")
779 dev.off()
780

```

```

781 print(xtable(cores.d.nonhazard %>% rownames_to_column() %>% select(
      rowname, cores.nonhazard, color.tikz) %>% rename("País" = "rowname",
      "$k$-cores" = "cores.nonhazard", "Color" = "color.tikz"),
782       digits = 0),
783       sanitize.colnames.function = bold,
784       booktabs = TRUE,
785       floating = TRUE,
786       file = sprintf("../LaTeX/tables/taula-%s-%s-%s-%s.tex", "
      coreness", "importacions", "nonhazard", "2018"))
787
788 # Díades
789 dyad_census(g.import.hazard.simple)
790 dyad_census(g.import.nonhazard.simple)
791
792 # Densitat
793 edge_density(g.import.hazard.simple)
794 edge_density(g.import.nonhazard.simple)
795
796 # Transitivitat/coeficient d'agrupació (clustering)
797 transitivity(g.import.hazard)
798 transitivity(g.import.nonhazard)
799
800 # Reciprocitat
801 reciprocity(g.import.hazard.simple, mode = "default")
802 reciprocity(g.import.hazard.simple, mode = "ratio")
803
804 reciprocity(g.import.nonhazard.simple, mode = "default")
805 reciprocity(g.import.nonhazard.simple, mode = "ratio")
806
807 # Clustering
808 g.import.hazard.un <- graph_from_data_frame(d = dades.hazard$vis.links
      , vertices = dades.hazard$vis.nodes, directed = FALSE)
809 g.import.hazard.un.simple <- simplify(g.import.hazard.un, remove.
      multiple = TRUE, remove.loops = TRUE)
810
811 kc <- cluster_fast_greedy(g.import.hazard.un.simple)
812 length(kc)
813 sizes(kc)
814 membership(kc)
815
816 memb.table <- data.frame(membership = as.matrix(membership(kc))) %>%
817   arrange(membership) %>%
818   rownames_to_column()
819
820 data.hazard <- left_join(x = memb.table, y = degree.table.hazard %>%
      select(-degree.in, -degree.out)) %>%
821   left_join(y = cores.d.hazard %>% rownames_to_column() %>% select(
      rowname, cores.hazard))
822
823 data.table.hazard <- data.hazard %>%
824   group_by(membership) %>%
825   summarise(sum.degrees = sum(degree),
826             sum.cores = sum(cores.hazard, na.rm = TRUE))
827
828 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "clustering", "
      importacions", "hazard", "2018"), width = 2000, height = 2000, res
      = 100)

```

```

829 plot(kc, g.import.hazard.un.simple)
830 dev.off()
831
832
833 ##
834
835 g.import.nonhazard.un <- graph_from_data_frame(d = dades.nonhazard$vis
  .links, vertices = dades.nonhazard$vis.nodes, directed = FALSE)
836 g.import.nonhazard.un.simple <- simplify(g.import.nonhazard.un, remove
  .multiple = TRUE, remove.loops = TRUE)
837
838 kc <- cluster_fast_greedy(g.import.nonhazard.un.simple)
839 length(kc)
840 sizes(kc)
841 membership(kc)
842
843 memb.table <- data.frame(membership = as.matrix(membership(kc))) %>%
844   arrange(membership) %>%
845   rownames_to_column()
846
847 data.nonhazard <- left_join(x = memb.table, y = degree.table.nonhazard
  %>% select(-degree.in, -degree.out)) %>%
848   left_join(y = cores.d.nonhazard %>% rownames_to_column() %>% select(
    rowname, cores.nonhazard))
849
850 data.table.nonhazard <- data.nonhazard %>%
851   group_by(membership) %>%
852   summarise(sum.degrees = sum(degree),
853             sum.cores = sum(cores.nonhazard))
854
855 png(sprintf("../LaTeX/figures/plot-%s-%s-%s-%s.png", "clustering", "
  importacions", "nonhazard", "2018"), width = 2000, height = 2000,
  res = 100)
856 plot(kc, g.import.nonhazard.un.simple)
857 dev.off()
858
859
860 # Assortativitat
861 assortativity_degree(g.import.hazard.simple)
862 assortativity_degree(g.import.nonhazard.simple)

```

Listing D.1: Codi per obtenir l'anàlisi descriptiu I i II de les característiques xarxes utilitzant com a paquet principal l'igraph

E | App web Shiny

```
1 # Paquets necessaris
2 library(plotly)
3
4 # Carreguem les dades des d'altres scripts
5 load("preprocessing.RData")
6
7 # Projectió del mapa
8 m <- list(
9   l = 0,
10  r = 0,
11  b = 0,
12  t = 0,
13  pad = 4
14 )
15
16 geo <- list(
17   showframe = TRUE,
18   showcoastlines = TRUE,
19   showocean = TRUE,
20   showland = TRUE,
21   showcountries = TRUE,
22   projection = list(
23     type = 'orthographic',
24     rotation = list(lon = -9, lat = 40, roll = 0)
25   ),
26   plotolution = "110",
27   landcolor = toRGB("gray95"),
28   countrycolor = toRGB("gray80"),
29   countrycolor = '#d1d1d1',
30   oceancolor = '#c9d2e0'
31 )
32
33
34
35 select_data <- function(var, any, pais){
36   if(var == 1){
37     dades <- import %>%
38       filter(if(any(pais == "Tots")) Year == any else Year == any &
39         From %in% pais) %>%
40       select(To) %>%
41       distinct()
42     return(list("dades" = dades))
43   }
44   if(var == 2){
45     dades <- export %>%
```

```

45     filter(if(any(pais == "Tots")) Year == any else Year == any & To
46             %in% pais) %>%
47     select(From) %>%
48     distinct()
49     return(list("dades" = dades))
50 }
51
52
53
54 map_plotly <- function(var, any, pais){
55
56     #####
57     # Importacions #
58     #####
59
60     if(var == 1){
61
62         sum_tonnes <- import %>%
63         filter(if(any(pais == "Tots")) Year == any else Year == any &
64                 From %in% pais) %>%
65         group_by(To) %>%
66         summarise(sum = sum(Quantity.in.tonnes_to),
67                   sum_cap = sum(Quantity.in.kg.per.capita_to)) %>%
68         arrange(To)
69
70         plot <- plot_geo() %>%
71         # From
72         add_markers(
73             data = import %>%
74             filter(if(any(pais == "Tots")) Year == any else Year == any
75                     & From %in% pais) %>%
76             group_by(From) %>%
77             select(From, start_long, start_lat) %>%
78             distinct(),
79             x = ~ start_long,
80             y = ~ start_lat,
81             text = ~ paste(sprintf("<b>%s</b>", From),
82                               sprintf("<em>Quantitat total:</em> %.2f tones",
83                                     sum(sum_tonnes$sum)),
84                               sep = "<br />"),
85             size = 5,
86             color = I("red"),
87             hoverinfo = "text",
88             alpha = 0.5,
89             name = ifelse(var == 1, "Pa\u00EFsos importadors", "Pa\u00EFsos exportadors")
90         ) %>%
91         # To
92         add_markers(
93             data = import %>%
94             filter(if(any(pais == "Tots")) Year == any else Year == any
95                     & From %in% pais) %>%
96             group_by(To) %>%
97             select(To, end_long, end_lat, Pop_to) %>%
98             distinct() %>%
99             arrange(To),

```

```

96     x = ~ end_long,
97     y = ~ end_lat,
98     text = ~ paste(sprintf("<b>%s</b>", To),
99                      sprintf("<em>Poblaci\u00F3: </em> %s habitants",
100                            format(Pop_to, big.mark = ",")),
101                      sprintf("<em>Quantitat: </em> %.2f tones", sum_
102                            tonnes$sum),
103                      sprintf("<em>Quantitat per capita: </em> %.4f kg
104                            ", sum_tonnes$sum_cap),
105                      sep = "<br />"),
106     size = 5,
107     color = I("blue"),
108     hoverinfo = "text",
109     alpha = 0.5,
110     name = ifelse(var == 1, "Pa\u00EFsos exportadors", "Pa\
111                u00EFsos importadors")
112 ) %>%
113 # Segments
114 add_segments(
115   data = import %>%
116     filter(if(any(pais == "Tots")) Year == any else Year == any
117           & From %in% pais),
118   x = ~ start_long, xend = ~ end_long,
119   y = ~ start_lat, yend = ~ end_lat,
120   size = I(1.5),
121   hoverinfo = "text",
122   alpha = 0.5,
123   hoverinfo = "none",
124   name = ifelse(var == 1, "Importacions", "Exportacions")
125 ) %>%
126 layout(gео = geo, showlegend = TRUE, legend = list(title = list(
127   text="<b>Llegenda</b>"), orientation = "h"), autosize =
128   FALSE, width = 600, height = 600, margin = m)
129
130 taula <- import %>%
131   filter(if(any(pais == "Tots")) Year == any else Year == any &
132     From %in% pais) %>%
133   group_by(To) %>%
134   select(To, Pop_to, Quantity.in.tonnes_to, Quantity.in.kg.per.
135     capita_to) %>%
136   summarise(sum = sum(Quantity.in.tonnes_to),
137             sum_cap = sum(Quantity.in.kg.per.capita_to),
138             pop = mean(Pop_to)) %>%
139   arrange(desc(sum)) %>%
140   rename("Pa\u00E9s exportador" = To, "Poblaci\u00F3" = pop, "
141     Quantitat en tones" = sum, "Quantitat per capita" = sum_cap)
142
143 return(list("plot" = plot, "taula" = taula))
144 }
145
146 #####
147 # Exportacions #
148 #####
149
150 if(var == 2){
151   sum_tonnes <- export %>%

```

```

143   filter(if(any(pais == "Tots")) Year == any else Year == any & To
        %in% pais) %>%
144   group_by(From) %>%
145   summarise(sum = sum(Quantity.in.tonnes_from),
146             sum_cap = sum(Quantity.in.kg.per.capita_from)) %>%
147   arrange(From)
148
149 plot <- plot_geo() %>%
150   # From
151   add_markers(
152     data = export %>%
153       filter(if(any(pais == "Tots")) Year == any else Year == any
154             & To %in% pais) %>%
155       group_by(From) %>%
156       select(From, start_long, start_lat, Pop_from) %>%
157       distinct(),
158     x = ~ start_long,
159     y = ~ start_lat,
160     text = ~ paste(sprintf("<b>%s</b>", From),
161                    sprintf("<em>Poblaci\u00F3: </em> %s habitants",
162                          format(Pop_from, big.mark = ",")),
163                    sprintf("<em>Quantitat:</em> %.2f tones", sum_
164                          tonnes$sum),
165                    sprintf("<em>Quantitat per capita:</em> %.4f kg
166                          ", sum_tonnes$sum_cap),
167                    sep = "<br />"),
168     size = 5,
169     color = I("red"),
170     hoverinfo = "text",
171     alpha = 0.5,
172     name = ifelse(var == 1, "Pa\u00EFsos exportadors", "Pa\
173               u00EFsos importadors")
174   ) %>%
175   # To
176   add_markers(
177     data = export %>%
178       filter(if(any(pais == "Tots")) Year == any else Year == any
179             & To %in% pais) %>%
180       group_by(To) %>%
181       select(To, end_long, end_lat) %>%
182       distinct(),
183     x = ~ end_long,
184     y = ~ end_lat,
185     text = ~ paste(sprintf("<b>%s</b>", To),
186                    sprintf("<em>Quantitat total:</em> %.2f tones",
187                          sum(sum_tonnes$sum)),
188                    sep = "<br />"),
189     size = 5,
190     color = I("blue"),
191     hoverinfo = "text",
192     alpha = 0.5,
193     name = ifelse(var == 1, "Pa\u00EFsos importadors", "Pa\
194               u00EFsos exportadors")
195   ) %>%
196   # Segments
197   add_segments(
198     data = export %>%

```

```

191     filter(if(any(pais == "Tots")) Year == any else Year == any
192             & To %in% pais),
193     x = ~ start_long, xend = ~ end_long,
194     y = ~ start_lat, yend = ~ end_lat,
195     size = I(1.5),
196     hoverinfo = "none",
197     alpha = 0.5,
198     name = ifelse(var == 1, "Importacions", "Exportacions")
199   ) %>%
200   layout(gео = geo, showlegend = TRUE, legend = list(title = list(
201     text = "<b>Llegenda</b>", orientation = "h"), autosize =
202     FALSE, width = 600, height = 600, margin = m)
203
204   taula <- export %>%
205     filter(if(any(pais == "Tots")) Year == any else Year == any & To
206            %in% pais) %>%
207     group_by(From) %>%
208     select(From, Pop_from, Quantity.in.tonnes_from, Quantity.in.kg.
209            per.capita_from) %>%
210     summarise(sum = sum(Quantity.in.tonnes_from),
211               sum_cap = sum(Quantity.in.kg.per.capita_from),
212               pop = mean(Pop_from)) %>%
213     ungroup() %>%
214     arrange(desc(sum)) %>%
215     rename("Pa\u00EDs exportador" = From, "Poblaci\u00F3" = pop, "
216            Quantitat en tones" = sum, "Quantitat per capita" = sum_cap)
217
218   return(list("plot" = plot, "taula" = taula))
219 }
220 }

```

Listing E.1: Codi per representar els enviaments transfronterers de residus sobre el mapamundi amb el paquet plotly

```

1 # Paquets necessaris
2 library(igraph)
3 library(visNetwork)
4 library(RColorBrewer)
5
6 network <- function(any, pais){
7   if(prod(dim(import %>% filter(if(any(pais == "Tots")) Year == any
8     else Year == any & From %in% pais))) == 0)
9     return(NULL)
10
11   else{
12     links <- import %>%
13       filter(if(any(pais == "Tots")) Year == any
14         else Year == any & From %in% pais) %>%
15       select(From, To, Quantity.in.tonnes_to, Quantity.in.kg.per.
16         capita_to, hazardous) %>%
17       rename(weight = Quantity.in.tonnes_to, from = From, to = To)
18
19     nodes <- data.frame(country = union(links$from, links$to)) %>%
20       left_join(y = import %>% filter(Year == any) %>% select(To, Pop_
21         to) %>% distinct(),
22         by = c("country" = "To")) %>%
23       rename(label = country)
24
25     vis.nodes <- nodes
26     vis.links <- links
27
28     #####
29     # Nodes #
30     #####
31
32     vis.nodes$id <- vis.nodes$label
33     vis.nodes$shadow <- FALSE
34     vis.nodes$borderWidth <- 2
35     vis.nodes$color.background <- ifelse(nodes$label %in% pais, "red",
36       "blue")
37     vis.nodes$color.border <- "black"
38     vis.nodes$group <- ifelse(nodes$label %in% pais, "Pa\u00EFsos
39       importadors", "Pa\u00EFsos exportadors")
40     vis.nodes$margin <- 1
41
42     #####
43     # Links #
44     #####
45
46     vis.links$id <- 1:nrow(vis.links)
47     vis.links$width <- links$weight/10000
48     vis.links$color <- ifelse(links$hazardous == 1, "#802621", "#1
49       b852e")
50     vis.links$arrows <- "from"
51     vis.links$smooth <- FALSE
52     vis.links$shadow <- FALSE
53
54     return(list("vis.links" = vis.links, "vis.nodes" = vis.nodes))
55
56

```

```
52   }  
53 }
```

Listing E.2: Codi per representar els enviaments transfronterers de residus segons teoria de xarxes amb el paquet `visNetwork`

```

1 # Paquets necessaris
2 library(shiny)
3 library(shinythemes)
4 library(shinyWidgets)
5 library(DT)
6 library(xtable)
7 library(tidyverse)
8
9 # Es carreguen les dades des d'altres scripts
10 source("plotly.R")
11 source("visNetwork.R")
12
13
14 ui <- tagList(
15   navbarPage("Anàlisi de dades mediambientals mitjançant teoria de
16     xarxes",
17     fluidPage(
18       p("Aquesta aplicació permet representar el conjunt de
19         dades d'enviaments transfronterers de residus,
20         originalment titulada ",
21         em("Transboundary Shipments of Waste"), ", extretes
22         del portal",
23         a("Eurostat", href = "https://ec.europa.eu/eurostat/
24           web/waste/data", target = "_blank"),
25         ".", br(), "Es mostren dues opcions de visualització.
26         La primera es troba en el panell", strong("
27           Mapamundi"),
28         ", on s'hi troba una representació de les dades
29         plasmades sobre el globus terraquí, així com
30         també una taula dinàmica en base a les dades
31         seleccionades.
32         La segona opció es troba en el panell", strong("Graf"
33         ), ", i dóna una representació visual sobre els
34         grafs, deixant triar quin tipus d'estructura es
35         desitja.
36         Ambdós panells es controlen mitjançant el menú de l'
37         esquerra, que serveix per a filtrar per categoria
38         de les dades (importacions o exportacions dels
39         residus),
40         any i país o països de procedència (en cas que s'hagi
41         triat veure les importacions) o destinació (en
42         cas que s'hagi seleccionat les exportacions).
43         Tanmateix, pel segon panell no s'hi representen les
44         exportacions de residus (per a més informació,
45         veure secció 2.4.4 de la memòria), però sí que es
46         diferencien els
47         residus perillosos (enllaços de color vermellós) dels
48         que no ho són (enllaços de color verdós). A més
49         a més, com més gruixut sigui aquest enllaç,
50         més quantitat de residus es transporten per a aquell
51         moviment.",
52         style = "text-align:justify;color:black;background-
53           color:#d1f1f1;padding:15px;border-radius:10px"
54       ),
55       theme = shinytheme("cosmo"),
56       titlePanel(""),
57       sidebarLayout(

```

```

33 sidebarPanel(
34   conditionalPanel(
35     "input.tab == 1",
36     prettyRadioButtons("import.export",
37       label = "Categoria",
38       choices = c("Importaciones", "
39         Exportaciones"),
40       shape = "round",
41       animation = "pulse",
42       plain = TRUE,
43       inline = TRUE,
44       outline = TRUE,
45       selected = "Importaciones",
46       icon = icon("check", style = "
47         color: #00CED1")
48     )
49   ),
50   conditionalPanel(
51     "input.tab == 2",
52     prettyRadioButtons("import.export",
53       label = "Categoria",
54       choices = "Importaciones",
55       shape = "round",
56       animation = "pulse",
57       plain = TRUE,
58       inline = TRUE,
59       outline = TRUE,
60       selected = "Importaciones",
61       icon = icon("check", style = "
62         color: #00CED1")
63     )
64   ),
65   setSliderColor("Transparent", 1),
66   chooseSliderSkin("Flat", color = "DarkTurquoise"),
67   sliderInput("year",
68     label = "Any",
69     min = min(waste$Year),
70     max = max(waste$Year),
71     value = ceiling((max(waste$Year) + min(
72       waste$Year))/2),
73     sep = "",
74     step = 1,
75     ticks = TRUE,
76     animate = TRUE
77   ),
78   selectizeInput("country1",
79     label = NULL,
80     choices = NULL,
81     multiple = TRUE,
82     options = list(plugins = list("
83       remove_button"),
84     placeholder = "
85       Selecciona un o
86       més països")
87   ),
88   conditionalPanel(
89     "input.tab == 2",

```

```

83     uiOutput("disposal.and.recovery.code"),
84     conditionalPanel(
85       'input.country1 != "" ',
86       prettyRadioButtons(inputId = "layout",
87         label = "Mode de
88           visualització",
89         choices = c("Arbre", "
90           Circular", "Estrella", "
91           Fruchterman i Reingol", "
92           Aleatori"),
93         shape = "round",
94         animation = "pulse",
95         plain = TRUE,
96         inline = FALSE,
97         outline = TRUE,
98         selected = "Arbre",
99         icon = icon("check", style =
100           "color: #00CED1")
101       )
102     ),
103     width = 2),
104 mainPanel(
105   tabsetPanel(id = "tab",
106     tabPanel(title = "Mapamundi", id = "
107       mapamundi", value = 1,
108       splitLayout(
109         plotlyOutput("plot"),
110         conditionalPanel(
111           'input.country1 != "" ',
112           div(dataTableOutput("table
113             "), style = "font-size
114               :80%"),
115           prettyRadioButtons("
116             downloadType", label =
117               "
118                 Selecciona
119                 un
120                 tipus
121                 de
122                 fitxer
123                 ",
124             choices
125               =
126               c("
127                 CSV
128                 " =
129                 ".
130                 csv
131                 ",
132                 "
133                 XLSX
134                 " =
135                 ".

```

```

111         xlsx
            ",
            "
            TeX
            " =
            ".
            tex
            "),
112     shape =
            "
            round
            ",
            animation
            =
            "
            pulse
            ",
113     plain =
            TRUE
            ,
114     inline
            =
            TRUE
            ,
115     outline
            =
            TRUE
            ,
116     icon =
            icon
            (
            "
            check
            ",
            style
            =
            "
            color
            :
            #00
            CED1
            ")
117     ),
118     downloadButton("
            downloadTable", "
            Descarregar taula")
119   ),
120   cellWidths = c("55%", "45%")
            ,
121   cellArgs = list(style = "
            height: 800px")
            )
122   ),
123   ),
124   ),
125   tabPanel(title = "Graf", id = "network"
            , value = 2,
126     br(),

```

```

127         span(textOutput("no_network"),
128               style = "color:red"),
129         visNetworkOutput("network",
130                           height = "1000px")
131       ),
132       width = 10
133     )
134   )
135 ),
136 )
137
138
139
140
141 server <- function(input, output, session){
142   #####
143   # Plotly #
144   #####
145
146   observe({
147     updateSelectizeInput(session,
148                           "country1",
149                           label = ifelse(input$import.export == "
150                                         Importacions", "Des de", "Fins a"),
151                           choices = c("Tots", sort(levels(world_coord$
152                                         country))),
153                           server = TRUE)
154   })
155
156   mapData <- reactive({
157     map_plotly(var = ifelse(input$import.export == "Importacions", 1,
158                             2), any = input$year, pais = input$country1)
159   })
160
161   output$plot <- renderPlotly({
162     mapData()$plot
163   })
164
165   output$table <- DT::renderDataTable({
166     mapData()$taula
167   },
168   options = list(
169     language = list(url = "//cdn.datatables.net/plug-ins/1.10.11/i18n/
170                      Catalan.json"),
171     pageLength = 15,
172     autowidth = TRUE,
173     scrollX = TRUE,
174     rownames = FALSE
175   )
176 )
177
178   output$downloadTable <- downloadHandler(
179     filename <- function(ext){
180       sprintf("taula-%s-%s-%s", tolower(input$import.export),
181              tolower(input$country1), input$year, input$downloadType)

```



```

177   },
178
179   content <- function(filename){
180     if(input$downloadType == ".csv"){
181       write.csv(mapData()$taula, filename, row.names = FALSE)
182     }
183
184     else if(input$downloadType == ".xlsx"){
185       write.xlsx(as.data.frame(mapData()$taula), filename, row.names
186                 = FALSE)
187     }
188
189     else if(input$downloadType == ".tex"){
190       bold <- function(x) {paste('\textbf{' , x , '}', sep = '')}
191       print(xtable(mapData()$taula,
192                    caption = sprintf("%s des de %s", input$import.
193                                     export, input$country1),
194                    label = sprintf("tab:%s-%s-%s", tolower(input$
195                                     import.export), tolower(input$country1),
196                                     input$year),
197                    math.style.exponents = "UTF-8"),
198            sanitize.colnames.function = bold,
199            include.rownames = FALSE,
200            booktabs = TRUE,
201            floating = TRUE,
202            file = filename)
203     }
204   }
205 )
206
207 #####
208 # Network #
209 #####
210
211 netData <- reactive({
212   if(!is.null(input$country1))
213     network(any = input$year, pais = input$country1)
214   else
215     NULL
216 })
217
218
219 output$network <- renderVisNetwork({
220   if(!is.null(netData()) & !is.null(input$country1)){
221     graf <- visNetwork(nodes = netData()$vis.nodes, edges = netData
222                        ()$vis.links) %>%
223       #visLegend() %>%
224       visOptions(highlightNearest = list(enabled = TRUE, hover =
225                                           TRUE, hideColor = "#f7f7f7"),
226                 nodesIdSelection = list(enabled = TRUE, main = "
227                                           Selecciona un país")) %>%
228
229       visExport(type = "png",
230                 label = paste0("Exporta com a png"))
231   }
232 })

```

```

227     if(input$layout == "Arbre")
228         visIgraphLayout(graf, layout = "layout_as_tree", randomSeed =
           123)
229
230     else if(input$layout == "Circular")
231         visIgraphLayout(graf, layout = "layout_in_circle", randomSeed
           = 123)
232
233     else if(input$layout == "Estrella")
234         visIgraphLayout(graf, layout = "layout_as_star", randomSeed =
           123)
235
236     else if(input$layout == "Fruchterman i Reingol")
237         visIgraphLayout(graf, layout = "layout_with_fr", randomSeed =
           123)
238
239     else if(input$layout == "Aleatori")
240         graf %>% visLayout(randomSeed = 123)
241
242     else
243         NULL
244 }
245
246 else
247     NULL
248 })
249
250 output$no_network <- renderText({
251     if(is.null(netData())$vis.nodes) & !is.null(input$country1))
252         sprintf("No hi ha registres en el dataset d'importacions per al
           país %s i l'any %d.", input$country1, input$year)
253     else
254         NULL
255 })
256
257 }
258
259 shinyApp(ui, server)

```

Listing E.3: Codi per elaborar l'app web Shiny

F | Informació de la sessió

```

R version 4.1.0 (2021-05-18)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 19041)

Matrix products: default

locale:
[1] LC_COLLATE=Spanish_Spain.1252  LC_CTYPE=Spanish_Spain.1252
[3] LC_MONETARY=Spanish_Spain.1252 LC_NUMERIC=C
[5] LC_TIME=Spanish_Spain.1252

attached base packages:
[1] stats      graphics  grDevices  utils      datasets  methods    base

other attached packages:
[1] plotly_4.9.4.1      DT_0.18          shinyWidgets_0.6.0
[4] shinythemes_1.2.0   shiny_1.6.0      sand_2.0.0
[7] igraphdata_1.0.1     RColorBrewer_1.1-2 visNetwork_2.0.9
[10] igraph_1.2.6         forcats_0.5.1    stringr_1.4.0
[13] dplyr_1.0.6          purrr_0.3.4      readr_1.4.0
[16] tidyr_1.1.3          tibble_3.1.2     ggplot2_3.3.5
[19] tidyverse_1.3.1     sna_2.6           statnet.common_4.5.0
[22] network_1.17.1      xtable_1.8-4

loaded via a namespace (and not attached):
[1] httr_1.4.2           sass_0.4.0         jsonlite_1.7.2      viridisLite
   _0.4.0
[5] modelr_0.1.8         bslib_0.2.5.1      assertthat_0.2.1    cellranger
   _1.1.0
[9] yaml_2.2.1           pillar_1.6.1       backports_1.2.1     lattice
   _0.20-44
[13] glue_1.4.2           digest_0.6.27      promises_1.2.0.1    rvest_1.0.0
[17] colorspace_2.0-2     htmltools_0.5.1.1  httpuv_1.6.1        pkgconfig
   _2.0.3
[21] broom_0.7.8          haven_2.4.1        scales_1.1.1         later_1.2.0
[25] generics_0.1.0       ellipsis_0.3.2     cachem_1.0.5         withr_2.4.2
[29] lazyeval_0.2.2       cli_3.0.0          magrittr_2.0.1       crayon
   _1.4.1
[33] readxl_1.3.1         mime_0.10          fs_1.5.0             fansi_0.5.0
[37] xml2_1.3.2           rsconnect_0.8.18   tools_4.1.0          data.table
   _1.14.0
[41] hms_1.1.0            lifecycle_1.0.0    munsell_0.5.0        reprex
   _2.0.0
[45] compiler_4.1.0       jquerylib_0.1.4    rlang_0.4.11         grid_4.1.0
[49] rstudioapi_0.13      htmlwidgets_1.5.3  crosstalk_1.1.1      gtable

```

```
      _0.3.0
[53] DBI_1.1.1      R6_2.5.0      lubridate_1.7.10 fastmap
      _1.1.0
[57] utf8_1.2.1     stringi_1.6.2  Rcpp_1.0.6      vctrs_0.3.8
[61] dbplyr_2.1.1   tidyselect_1.1.1 coda_0.19-4
```

Listing F.1: Informació de la sessió



This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International license.