



## Valoració d'habitatges automatitzada

**Nom Estudiant:** Goran Polic Maric

Estudiant de Màster en Enginyeria Informàtica

Àrea: Intel·ligència Artificial

**Nom Consultor/a:** Samir Kanaan Izquierdo

**Nom Professor/a responsable de l'assignatura:** Carles Ventura Royo

Data lliurament memòria final: 28 de desembre



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

**B) GNU Free Documentation License (GNU FDL)**

Copyright © ANY 2016/2017 Goran Polic

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

### **C) Copyright**

© (l'autor/a) Goran Polic

Reservats tots els drets. Està prohibit la reproducció total o parcial d'aquesta obra per qualsevol mitjà o procediment, compresos la impressió, la reprografia, el microfilm, el tractament informàtic o qualsevol altre sistema, així com la distribució d'exemplars mitjançant lloguer i préstec, sense l'autorització escrita de l'autor o dels límits que autoritzi la Llei de Propietat Intel·lectual.

## FITXA DEL TREBALL FINAL

<b>Títol del treball:</b>	<i>Valoració automatitzada d'habitatges</i>
<b>Nom de l'autor:</b>	<i>Goran Polic Maric</i>
<b>Nom del consultor/a:</b>	Samir Kanaan Izquierdo
<b>Nom del PRA:</b>	Carles Ventura Royo
<b>Data de lliurament (mm/aaaa):</b>	<i>28 de desembre</i>
<b>Titulació o programa:</b>	<i>Màster Universitari en Enginyeria Informàtica</i>
<b>Àrea del Treball Final:</b>	<i>Intel·ligència Artificial</i>
<b>Idioma del treball:</b>	<i>Català</i>
<b>Paraules clau</b>	<i>Valoració, habitatges, immobiliària</i>
<p><b>Resum del Treball (màxim 250 paraules):</b> <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i></p>	
<p>Aquest projecte s'ha volgut fer per poder automatitzar valoracions d'habitatges i poder comparar aquests resultats obtinguts amb els preus que hi marquen els seus comercials. És un tema rellevant perquè alguns habitatges es valoren malament (sobretot són més cars del normal) i això fa que els mateixos no es venguin o portin més temps a la cartera de la immobiliària.</p> <p>El problema s'ha volgut tractar amb la intel·ligència artificial per calcular els preus més aproximats dels habitatges en funció del seu districte, barri, estat, metres quadrats, número d'habitacions i els habitatges venuts que tenen les mateixes característiques o semblances als que volem valorar.</p> <p>El projecte s'ha centrat en 3 punts:</p> <p>Estadístiques, que és un pas previ en Intel·ligència Artificial que ens dóna una certa idea sobre les propietats: aquelles amb més comoditats i també més cares, amb menys comoditats i també més barates, els districtes amb més ascensors, balcons, garatges, habitatges més antics, etc</p> <p>El segon punt important són els Kmeans on hi ha dades significatives després d'observar com s'aglutinen les dades al voltant dels centroides. Les propietats que es venen millor són en la seva totalitat pisos. Els pisos que es venen millor tenen un número d'habitacions i de comoditats i més terrasses.</p>	

Arbres de decisió és el tercer punt. Els resultats del classificador d'habitatges a partir dels arbres de decisió ha estat molt decebedor. Creiem que hi ha gran disparitat en número d'habitatges a cada valor de la classe. La mostra d'elements de cada valor de classe són molt variables. Per exemple en un districte, la majoria d'habitatges (el 80 per cent) es venen amb facilitat i en canvi hi ha pocs individus a la resta de valors de classe.

El propòsit final d'aquest projecte ha estat el fet de poder fer servir el Python, Tkinter, Flask i Javascript per obtenir resultats gràfics que són més "user friendly" per a l'usuari final.

**Abstract (in English, 250 words or less):**

THE OBJECTIVE OF THIS PROJECT IS TO DISCOVER CERTAIN PATTERNS AND TENDENCIES IN CUSTOMER PREFERENCES IN PURCHASING A PROPERTY, AND TO FIND OUT WHY THEY WISH TO PURCHASE A PROPERTY IN A CERTAIN NEIGHBOURHOOD AND NOT IN ANOTHER. THE TARGET AREA FOR THE STUDY IN THE CITY OF BARCELONA, WHICH IS DIVIDED INTO DISTRICTS AND LATER INTO NEIGHBOURHOODS CALLED "BARRIS".

THE FIRST CHALLENGE WAS TO FIND THE PROPER DATABASE. AFTER MANY TRIALS AND ADVICE GIVEN BY AGENTS AND REAL ESTATE PROFESSIONALS, THE AUTHOR CHOSE TO CLASSIFY THE PROPERTIES SIMPLY BY DISTRICTS ALONE, AS THE LIMITED NUMBER OF PROPERTIES WOULD OFFER MORE ACCURATE RESULTS IN THIS WAY.

THE PROBLEM WITH THE DATABASE IS DUE TO THE CONSIDERABLE DIFFICULTY IN OBTAINING ENOUGH TRAINING DATA IN THE CLASS "EASY TO SELL", SINCE IT IS NECESSARY TO CONTRAST THE DIFFERENCE BETWEEN WHEN THE PROPERTY FIRST CAME ON THE MARKET AND WHEN IT WAS ACTUALLY SOLD. HOW EASY IT IS TO SELL A PROPERTY WAS DETERMINED THROUGH COMMON SENSE AND LOGIC.

IN SPITE OF THIS, IT WAS POSSIBLE TO REACH SOME CONCLUSIONS AFTER ANALYSIS OF THE K-MEANS ALGORITHM RESULTS.

IN FACT, THE MOST IMPORTANT GOAL OF THIS PROJECT HAS BEEN ACHIEVED, DUE TO THE FACT THAT PROGRAMMING TECHNOLOGIES LIKE PYTHON TKINTER, PYTHON FLASK, AND JAVASCRIPT WERE USED, IN ORDER TO OBTAIN GRAPHIC RESULTS THAT ARE MORE APPEALING TO CUSTOMERS OR ANYONE ELSE INTERESTED IN THESE FINDINGS.

THIS IS JUST THE STARTING POINT. THE AUTHOR HOPES TO GO DEEPER INTO THIS FIELD IN THE NEAR FUTURE.

# ÍNDEX

ÍNDEX .....	6
ÍNDEX DE LES FIGURES.....	8
1. Introducció.....	11
1.1. Context i justificació del Treball .....	11
1.2. Objectius del Treball.....	12
1.3. Enfocament i mètode seguit .....	13
1.4. Planificació del Treball .....	20
1.5. Breu sumari de productes obtinguts.....	24
1.6. Breu descripció dels altres capítols de la memòria.....	24
2. Dades de l'estudi d'habitatges .....	25
2.1. Anàlisi de les dades que es tenen : xlsx d'habitatges .....	25
2.1.1. Camps numèrics: discrets i continus .....	28
2.1.2. Atributs qualitatiu .....	28
2.2. Tractament de dades previ a l'anàlisi .....	28
2.3. K-Means ampliat .....	29
3. Estudi estadístic de les dades .....	40
3.1. Atributs i paràmetres estadístics interessants.....	40
3.2. Gràfics de les dades: Histogràmes, etcètera .....	42
3.3. Conclusions de les dades.....	46
4. Classificador de les dades. Arbres de decisió .....	48
4.1. Aplicació de Filtre pisos venuts i no venuts .....	48
4.2. Definició de facilitat de venda d'habitatge .....	48
4.3. Classes i Atributs .....	48
4.4. Funció Goodness .....	49
4.5. Arbre de decisió .....	50
4.6. Conclusions de l'arbre de decisió.....	50
5. Dades de clients de la nostra base de dades d'habitatges.....	53
5.1. Anàlisi de les dades que es tenen : xlsx de clients .....	53
5.1.1. Camps numèrics: discrets i continus .....	53
5.1.2. Atributs qualitatiu .....	53

5.2. Tractament de dades previ a l'anàlisi .....	54
5.3. Tipus de clients. Anàlisi K-means clustering .....	54
6. Recomanació pels nostres clients.....	55
6.1. Filtre de pisos no venuts .....	55
6.2. Funció similitud entre habitatges.....	56
6.3. Implementació d'un recommender .....	56
7. Creació d'un lloc web local .....	58
7.1. Django vs flask.....	58
7.2. Target del lloc web .....	65
7.3. Esquema del lloc web i usabilitat .....	66
7.4. Interrelació del lloc web amb codi python implementat .....	66
7.5. Interpretació del lloc web .....	66
8. Adaptació del lloc web per utilitzar-lo a internet.....	67
8.1. Integració a diferents navegadors .....	67
9. Millores i futures actualitzacions.....	68
10. Pressupost del projecte .....	69
11. Conclusions.....	70
12. Glossari .....	71
13. Bibliografia.....	72
14. Annexos .....	74
14.1. Codi font arxiu MAIN.py.....	74
14.2. Codi font arxiu DTnum.py .....	80
14.3. Codi font arxiu estadistiques-tkinter.py.....	81

## ÍNDIX DE LES FIGURES

Figura 1 - Imatge - Casa - Font .....	11
Figura 2 - Valoració habitatges - Font.....	12
Figura 3 - Intel·ligència Artificial - Font.....	15
Figura 4 - Estat dels habitatges.....	16
Figura 5 - Districtes de Barcelona.....	16
Figura 6 - Barris del districte Ciutat Vella .....	16
Figura 7 - Tipus d'habitatges.....	16
Figura 8 - Barris del districte Sant Martí.....	17
Figura 9 - Barris del districte Nou Barris.....	17
Figura 10 - Barris del districte Sant Andreu.....	18
Figura 11 - Barris del districte Sants - Montjuïc .....	18
Figura 12 - Barris del districte Sarrià - Sant Gervasi .....	18
Figura 13 - Barris del districte Les Corts .....	18
Figura 14 - Barris del districte Gràcia .....	19
Figura 15 - Barris del districte Eixample .....	19
Figura 16 - Barris del districte Horta - Guinardó .....	19
Figura 17 - Taula dels timings .....	21
Figura 18 - Activitats resumides .....	22
Figura 19 - Diagrama de Gannt - Timings .....	23
Figura 20 - Resultats del Kmeans de Les Corts .....	30
Figura 21 - Resultats del Kmeans de Ciutat Vella .....	31
Figura 22 - Resultats del Kmeans d'Eixample.....	33
Figura 23 - Resultats del Kmeans de Gràcia .....	34
Figura 24 - Resultats del Kmeans de Sant Andreu.....	36
Figura 25 - Resultats del Kmeans de Sarrià - Sant Gervasi .....	37
Figura 26 - Resultats del Kmeans de Sants – Montjuïc .....	38
Figura 27 - Mòdul Statistics .....	40
Figura 28 – Histograma – Ciutat Vella .....	42



Figura 29 - Histograma - Sant Gervasi .....	43
Figura 30 - Histograma - Eixample.....	43
Figura 31 - Histograma - Les Corts.....	44
Figura 32 - Histograma - Sants Montjuïc .....	44
Figura 33 - Histograma - Gràcia .....	45
Figura 34 - Histograma - Sant Martí .....	45
Figura 35 - Arbres de decisió de cada districte .....	49
Figura 36 - Funció Goodness .....	50
Figura 37 - Pisos recomanats.....	55
Figura 38 – Recommender .....	57
Figura 39 – Filtres de búsqueda de pisos .....	57
Figura 40 - Logo de Django - Font.....	58
Figura 41 - Logo de Flask - Font .....	58
Figura 42 - Decorators .....	59
Figura 43 – Esquema de la web .....	65
Figura 44 – Navegadors – Font.....	67

Eines de desenvolupament preparades i provades

APARTAT DE L'ÍNDEX	ENTORN PYTHON I MÒDULS
1. DADES DE L'ESTUDI D'HABITATGES	Anaconda 4.2.0 64 bits Python 3.5 numpy sklearn.preprocessing sklearn.decomposition.PCA sklearn.manifold.MDS matplotlib.pyplot pylab
2. ESTUDI ESTADÍSTIC DE LES DADES 3. CLASSIFICADOR DE LES DADES. ARBRES DE DECISIÓ	Anaconda 4.2.0 64 bits Python 3.5 numpy matplotlib.pyplot math statistics DTnum (@author: Andrés Cencerrado)
4. DADES DE CLIENTS DE LA NOSTRA BASE DE DADES D'HABITATGES	Anaconda 4.2.0 64 bits Python 3.5  matplotlib.pyplot numpy sklearn.preprocessing sklearn.decomposition.PCA sklearn.manifold.MDS sklearn.cluster.KMeans sklearn.preprocessing sklearn.decomposition.PCA sklearn.manifold.MDS
5. RECOMANACIÓ PELS NOSTRES CLIENTS	Anaconda 4.2.0 64 bits Python 3.5 operator
6. CREACIÓ D'UN LLOC WEB LOCAL	Definir Django o Flask
7. ADAPTACIÓ DEL LLOC WEB PER UTILITZAR-LO A INTERNET	Definir Django o Flask

# 1. INTRODUCCIÓ

## 1.1. CONTEXT I JUSTIFICACIÓ DEL TREBALL

Una immobiliària multinacional, amb vàries oficines a Barcelona, ens demana automatitzar les seves valoracions d'habitatges i poder comparar aquests resultats obtinguts amb els preus que hi marquen els seus comercials. És un tema rellevant perquè alguns habitatges es valoren malament (sobretot són més cars del normal) i això fa que els mateixos no es venguin o portin més temps a la cartera de la immobiliària.

El problema es podria resoldre o almenys millorar fent servir la intel·ligència artificial per calcular els preus més aproximats dels habitatges en funció del seu districte, barri, estat, metres quadrats, número d'habitacions i els habitatges venuts que tenen les mateixes característiques o semblances als que volem valorar.

Amb la valoració automatitzada es pot aconseguir un preu més correcte i també comparar-ho amb totes els preus assignats als habitatges fins ara per poder corregir-los, si cal.



Figura 1 - Imatge - Casa - [Font](#)

## 1.2. OBJECTIUS DEL TREBALL

La immobiliària té actualment unes 8000 propietats en venda a la seva cartera i algunes d'elles tenen els preus incorrectes segons el mercat i l'històric de vendes de propietats similars.

**Els objectiu d'aquest projecte són:**

- Facilitar la tasca als comercials immobiliaris alhora de valorar els habitatges
- Ajudar als comercials nous a valorar habitatge perquè no tenen tanta experiència
- Ajudar a la immobiliària a revisar tot el seu stock i comprovar quines propietats s'han valorat correctament i quines no de les 8000 que té actualment a la cartera
- Augmentar el número de leads (contactes obtinguts a la seva pàgina web i portals immobiliaris)
- Optimitzar el treball i temps invertit en la gestió de les propietats
- Obtenir més beneficis
- Millor reputació com a empresa



Figura 2 - Valoració habitatges - [Font](#)

### 1.3. ENFOCAMENT I MÈTODE SEGUIT

La estratègia és usar la intel·ligència artificial i en concret el llenguatge Python per poder determinar els preus de les propietats i comparar-los amb els preus actuals.

El producte desenvoluparà des de zero, però alguns algorismes s'han reutilitzat d'altres projectes o proves fets en l'assignatura Intel·ligència Artificial Avançada.

Potser aquesta estratègia és la més apropiada perquè tindrà més inputs que s'hauran de tenir en compte i poder obtenir uns preus més correctes i justos.

Els inputs emprats podrien ser, per exemple:

- ◆ Tipus de propietat (habitatge):

- Casa:
  - Unifamiliar
  - Adossada
  - Villa
  - Pareada
- Pis:
  - Apartament
  - Àtic
  - Dúplex
  - Loft

- ◆ Estat de l'habitatge:

- Reformat
- Per reformar
- Obra nova

- ◆ Districte: (més endavant es detallaran els barris d'aquests districtes)
  - Ciutat Vella:
  - Eixample:
  - Sants-Montjuïc:
  - Les Corts:
  - Sarrià-San Gervasi:
  - Gràcia:
  - Horta-Guinardó:
  - Nou Barris:
  - San Andreu:
  - San Martí:
- ◆ Codi postal
- ◆ Any de construcció:
- ◆ Mobles:
  - Moblat
  - No moblat
- ◆ Preu:
- ◆ Preu metre quadrat:
- ◆ Superfície construïda(m<sup>2</sup>):
- ◆ Localitat:
- ◆ Habitacions:
- ◆ Dormitoris:
- ◆ Banys:
- ◆ Aire condicionat (Sí / No)
- ◆ Terrassa (Sí / No)
- ◆ Balcó (Sí / No)
- ◆ Places garatge incloses:
- ◆ Sauna: (Sí / No)
- ◆ Gimnàs: (Sí / No)
- ◆ Piscina: (Sí / No)

- ◆ Tipus de calefacció:
- ◆ Ascensor: (Sí / No)
- ◆ Certificació d'eficiència energètica (CEE): (Sí / No)

Introduint aquests dades, obtindrem preus dels habitatges.

Un cop tinguem els preus dels habitatges, podem comparar els preus de les propietats existents de la immobiliària.

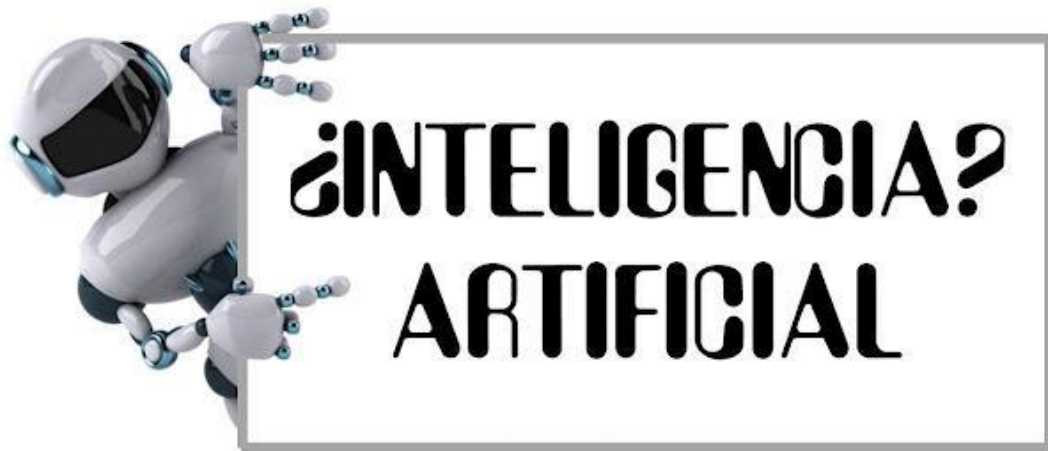


Figura 3 - Intel·ligència Artificial - [Font](#)

Per tant, en obrir l'aplicatiu, podrà consultar unes estadístiques per barris, districtes, superfície, número d'habitacions, l'estat de l'habitatge, etc, el programa ens donarà el preu de l'habitatge de les propietats que es trobin en diferents punts de Barcelona (districte + barri). Tots aquests factors seran claus per determinar el preu i recomanació d'immobles similars o que tinguin les mateixes característiques en diferents barris de Barcelona.

A continuació farem uns esquemes de com quedarà la classificació del districtes i barris.

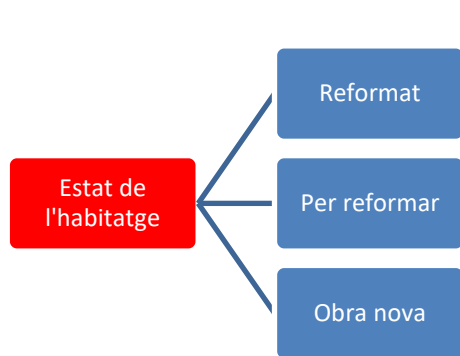


Figura 4 - Estat dels habitatges

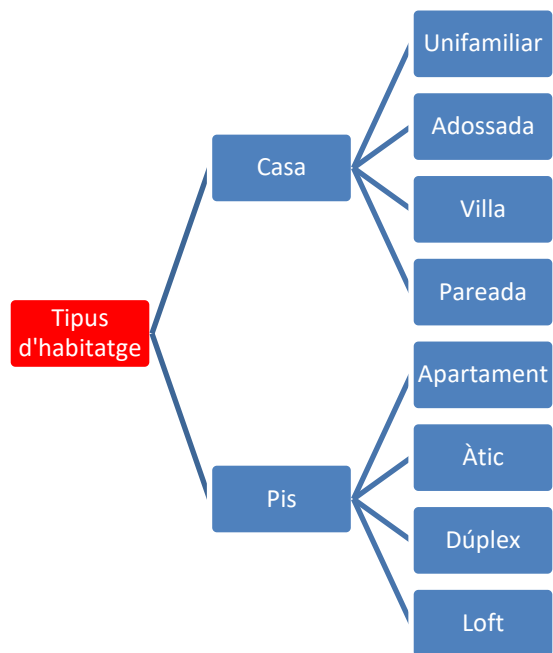


Figura 7 - Tipus d'habitatges

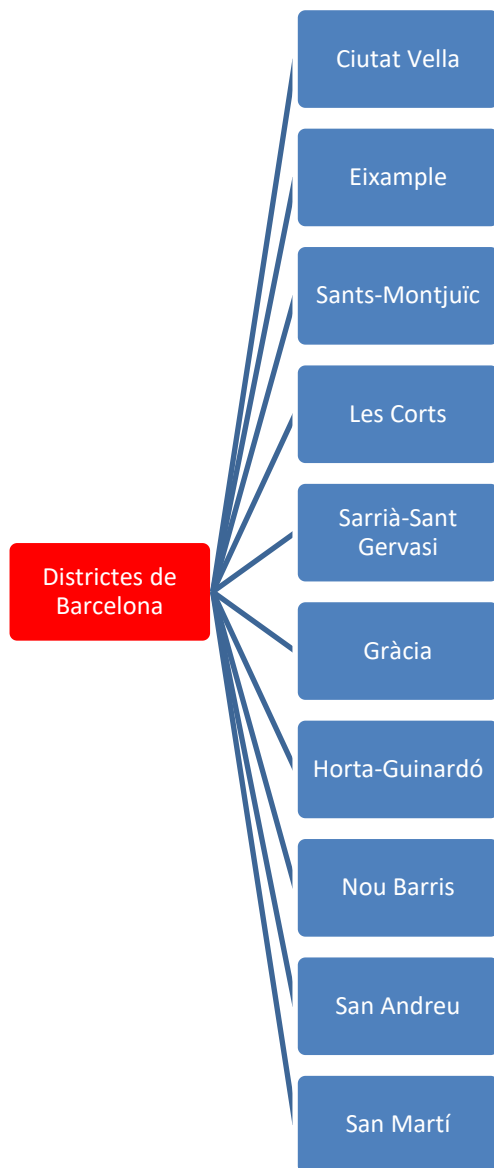


Figura 5 - Districtes de Barcelona

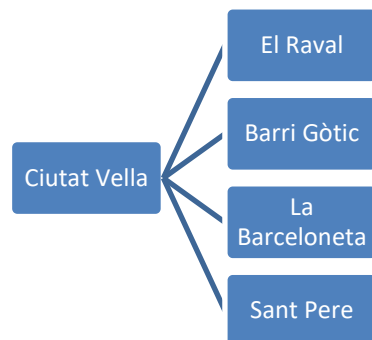


Figura 6 - Barris del districte Ciutat Vella



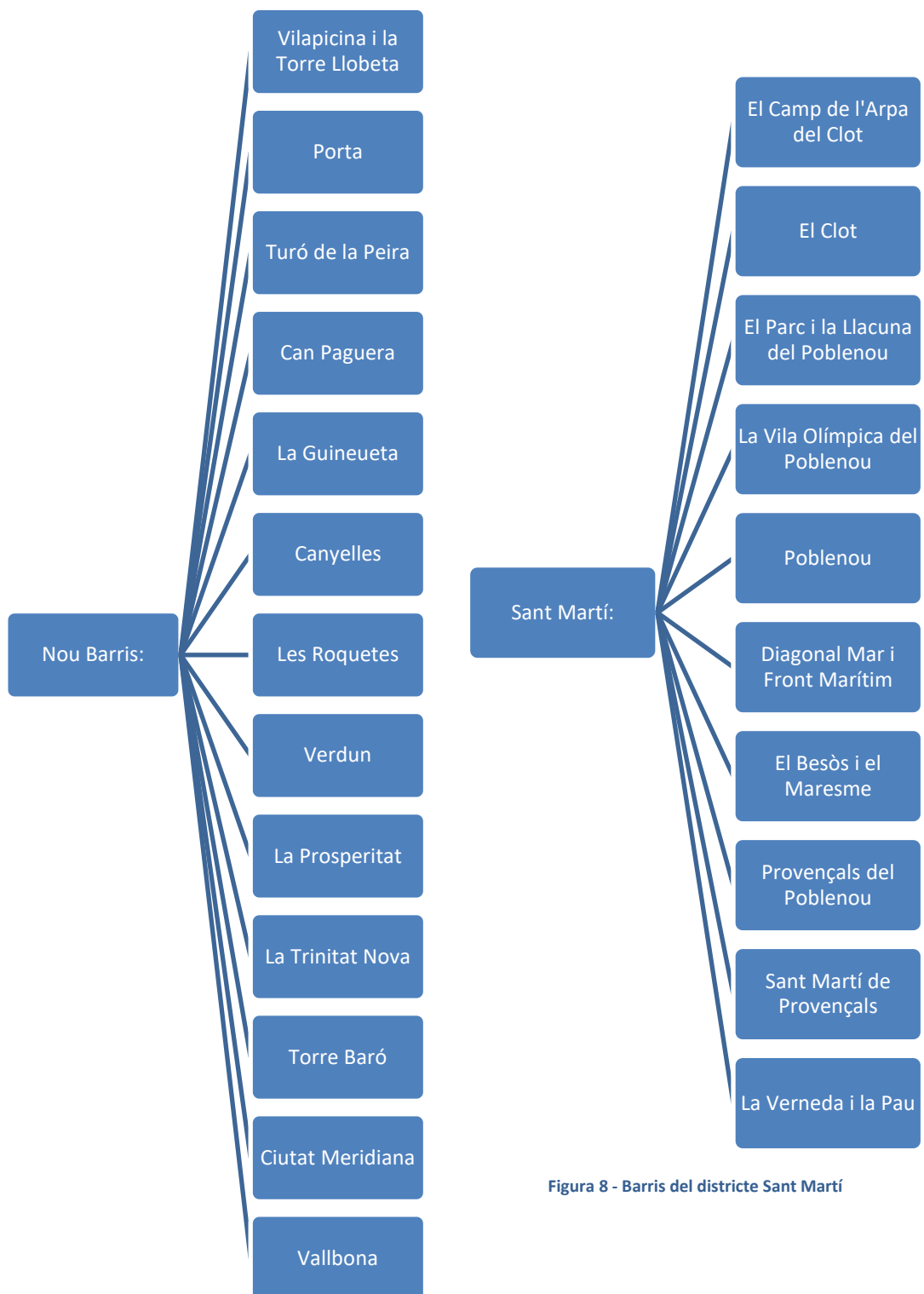


Figura 9 - Barris del districte Nou Barris

Figura 8 - Barris del districte Sant Martí

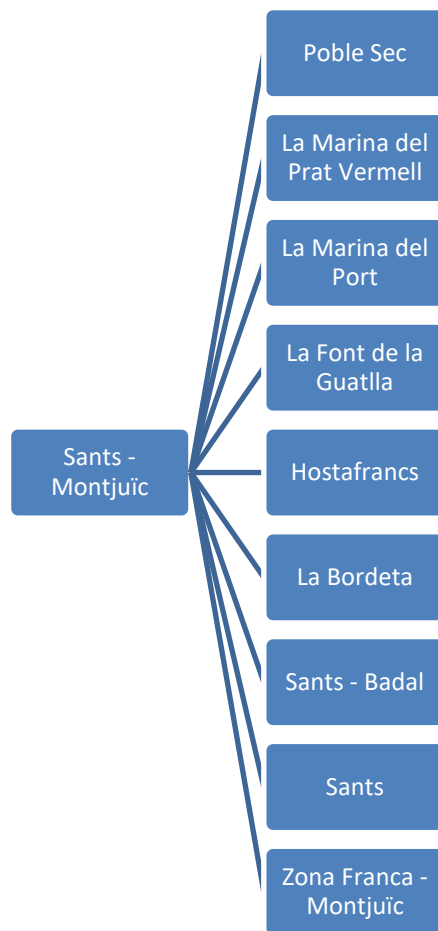


Figura 11 - Barris del districte Sants - Montjuïc

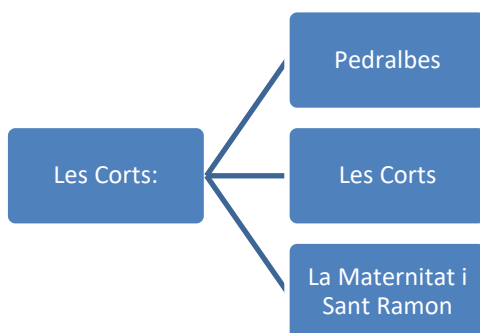


Figura 13 - Barris del districte Les Corts

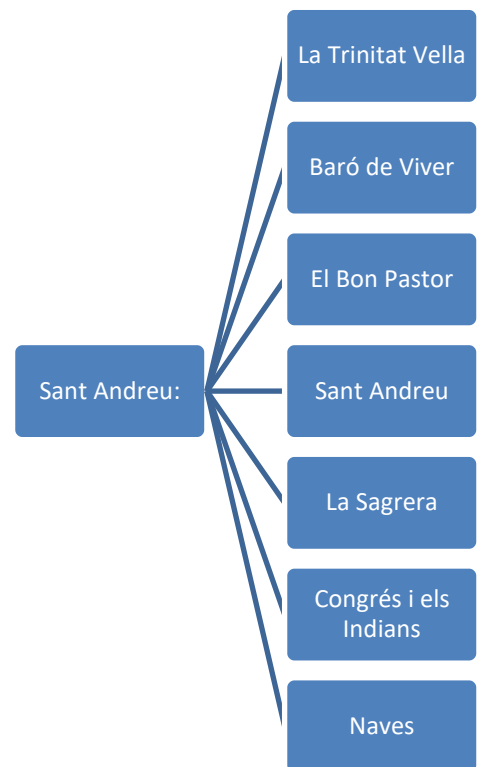


Figura 10 - Barris del districte Sant Andreu

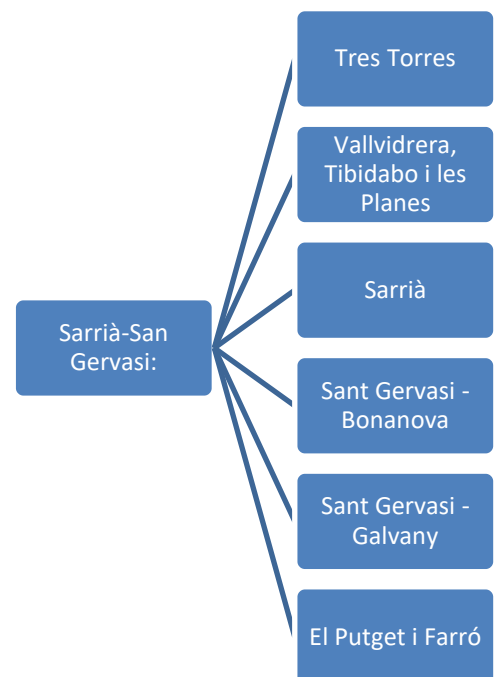


Figura 12 - Barris del districte Sarrià - Sant Gervasi

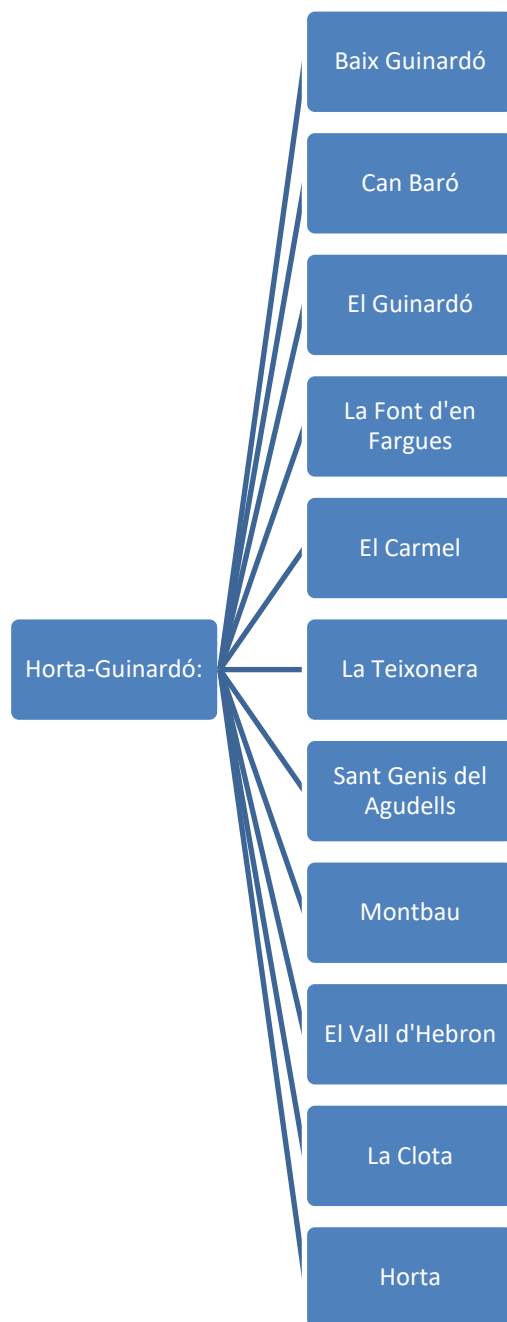


Figura 16 - Barris del districte Horta - Guinardó

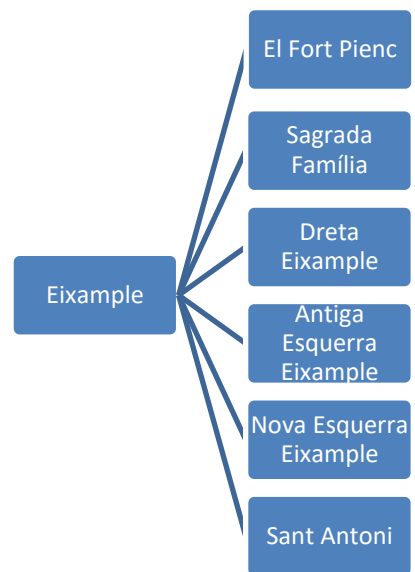


Figura 15 - Barris del districte Eixample

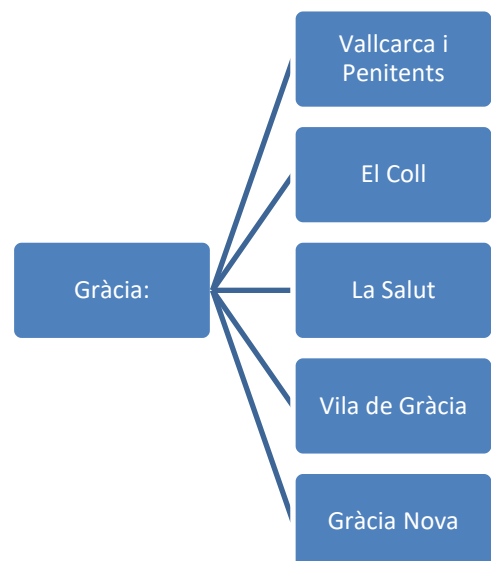


Figura 14 - Barris del districte Gràcia

#### 1.4. PLANIFICACIÓ DEL TREBALL

A continuació detallem els timings de la planificació des d'ara al lliurament del projecte

Activitat	Data	Activitat	Esdeveniment
1	21-27 de setembre	Definir els objectius del projecte i l'abast	
2	28 de setembre a 4 d'octubre	Definir les diferents tasques. Realitzar la planificació del projecte	
3	5 d'octubre	Proposta inicial del projecte, amb objectius, característiques del problema, propostes de solució, planificació i breu revisió de l'estat de l'art i del potencial mercat si s'escau.	<b>Lliurament pla de treball PAC1</b>
4	6 – 15 d'octubre	Crear un arxiu d'Excel que contindrà unes 8000 propietats que té la immobiliària a la seva cartera definint totes les opcions establertes a l'apartat 1.3. d'aquest mateix document. Començar a configurar l'entorn de treball de Python.	
5	16 d'octubre fins al 2 de novembre	Preparació de la PAC 2 Memòria: document amb l'estructura (capítols, seccions, no cal redactar-ho tot). Preparar les eines, tècniques i dades. Mostrar resultats senzills, amb dades de prova. Seguir codificant l'aplicatiu a Python. Lliurament de la PAC el 2 de novembre	<b>Lliurament la memòria PAC2</b>
6	Del 3 al 25 de	Codificant	

	novembre		
7	Del 26 al 29 de novembre	Codificant + Redacció dels apartats de la memòria	
8	30 de novembre	Versió "beta" de la memòria final i resultats del treball de codificació amb Python	<b>Lliurament la memòria PAC3</b>
9	1-18 de desembre	Revisions, comprovació dels resultats, testing del codi	
10	19 - 23 de desembre	Debat / Conclusions final, glossari i bibliografia	<b>Lliurament presentació provisional</b>
11	24-27 de desembre	Debat / Conclusions final, glossari i bibliografia	
12	28 de desembre	Memòria final i presentació del treball en vídeo (diapositives amb veu).	<b>Lliurament memòria completa amb les conclusions PAC4</b>

Figura 17 - Taula dels timings

Activitat	Inici	Finalització	Durada (dies)
Activitat 1	21/09/2016	27/09/2016	6
Activitat 2	28/09/2016	04/10/2016	6
Activitat 3	05/10/2016	05/10/2016	0
Activitat 4	06/10/2016	15/10/2016	9
Activitat 5	16/10/2016	02/11/2016	17
Activitat 6	03/11/2016	25/11/2016	22
Activitat 7	26/11/2016	29/11/2016	3
Activitat 8	30/11/2016	30/11/2016	0
Activitat 9	01/12/2016	18/12/2016	17
Activitat 10	19/12/2016	23/12/2016	4
Activitat 11	24/12/2016	26/12/2016	2
Activitat 12	27/12/2016	28/12/2016	1

Figura 18 - Activitats resumides

I també farem el Diagrama de Grannt per dividir el projecte en diferents etapes juntament amb les dades claus d'execució dels treball i lliurament dels mateixos.

El projecte va començar el dia 27 de setembre i la seva finalització i lliurament serà el dia 28 de desembre.

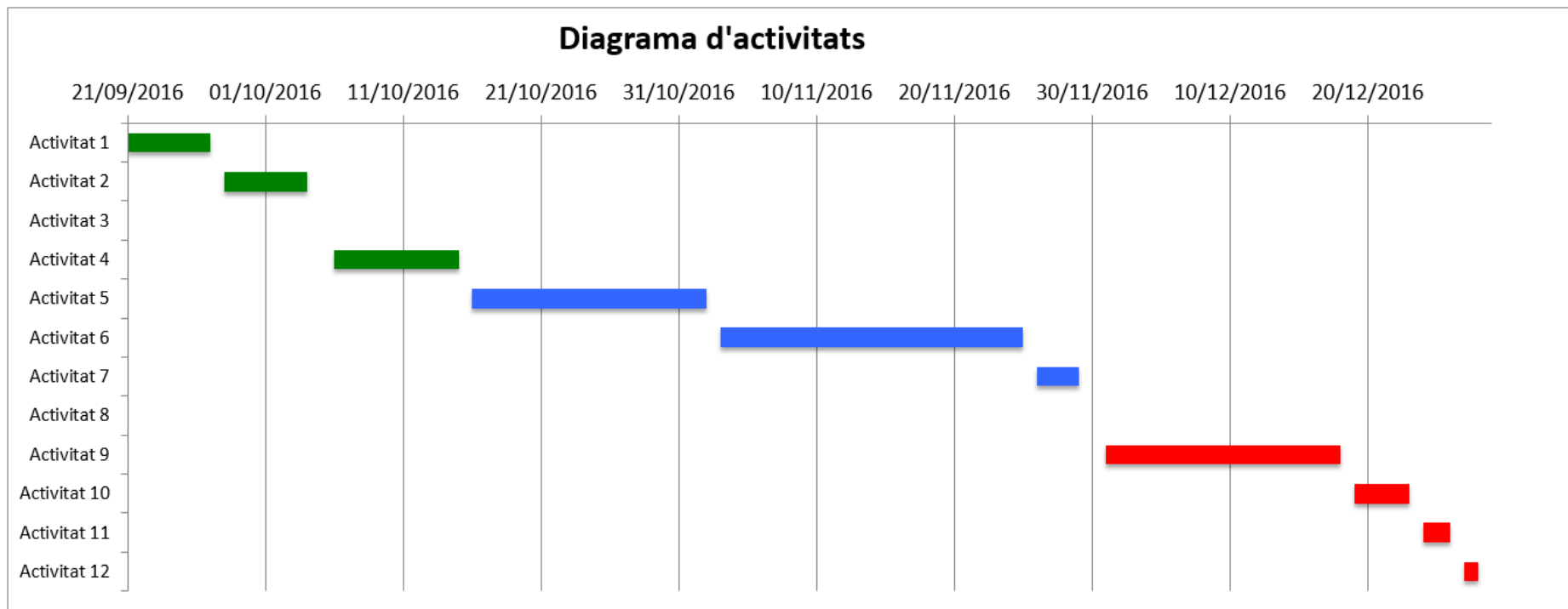


Figura 19 - Diagrama de Gannt - Timings

## 1.5. BREU SUMARI DE PRODUCTES OBTINGUTS

El producte obtingut serà un aplicatiu que per un costat ens determinarà els preus de les propietats i per un altre, farà una comparació del preus obtinguts amb els actuals de la immobiliària.

Sabrem en quins districtes tenim les propietats més cares, més barates, aquelles zones on hi ha més garatges, on els habitatges són més vells o nous, propietats més moblades, amb més habitacions, més terrasses i aire condicionat, etc.

Posteriorment es recomanaran habitatges similars segons les característiques que hem establert: preu, districte, superfície, número d'habitacions, etc.

## 1.6. BREU DESCRIPCIÓ DELS ALTRES CAPÍTOLS DE LA MEMÒRIA

A partir d'aquest apartat començarem amb les dades d'estudi d'habitatges analitzant les dades, els seu tractament previ i després fer un estudi dels K-means ampliats.

Posteriorment, al segon capítol farem un estudi estadístic de les dades amb els gràfics corresponents i conclusions.

Més tard parlarem de la classificació de les dades, és a dir, dels arbres de decisió. Filtrarem els pisos venuts i no venuts, la seva facilitat de venda. Farem servir la funció Goodness i al final unes conclusions dels arbres de decisió.

Posteriorment es farà una recomanació pels nostres clients, filtrar els pisos venuts i no venuts, implementar la funció de similitud entre diferents habitatges, implementació d'un "recommender".

El pas següent és la creació d'un lloc web per poder fer servir l'aplicació d'una manera més senzilla. I després farem l'adaptació de la web per utilitzar-la online.



## 2. DADES DE L'ESTUDI D'HABITATGES

### 2.1. ANÀLISI DE LES DADES QUE ES TENEN : XLSX D'HABITATGES

El primer que cal fer és analitzar les variables que defineixen cadascun dels camps per veure els valors que prenen i com el treballarem posteriorment.

Els camps de la base de dades són 31 i alguns d'ells seran tractats de manera especial i altres són obsolets o no afegeixen res important.

Tenim en total 8416 entrades amb diferent informació.

**ID:** L'identificador (clau primària). Valor únic que identifica cadascun dels pisos en venda. Pren valors entre 1 i 8416.

**Tipus d'habitatge:** Pis o casa → binari

**Pis assignat a:** Ens diu el comercial que porta la venda de la propietat. Per exemple, comercials de l'1 al 21.

**Subtipus:** El pis pot ser: loft, dúplex, apartament, etc. La casa pot ser: unifamiliar, villa, adossada, pareada

**Estat:** Reformat, no reformat i obra nova → Literal

**Propietari:** Aquí tenim la informació dels propietaris. S'han utilitzat els noms: propietari 1, propietari 2, etc per no fer servir els noms reals

**Districte:** Aquí tenim els 10 districtes de Barcelona → Literal

**Barri:** Cada districte està compost de diferents barris. Aquí tenim la llista de tots ells.  
→ Literal

**Codi postal** dels diferents barris de Barcelona → Literal

**Any de construcció:** Indica l'any que es va construir l'habitatge → Interval numèric

**Mobles:** Moblat o no moblat → Binari

El **preu** actual del moble o el preu amb que es va vendre l'immoble → Numèric

**Preu per metre quadrat:** Preu dels habitatges per metre quadrat. Aquesta xifra varia per barris → Numèric

**Superfície dels habitatges** en m2 → Numèric

**Localitat** que en el nostre cas és sempre Barcelona ja que el projecte està centrat només en aquesta ciutat → Literal

**Habitacions:** El número d'habitació que té l'habitatge → Numèric

**Dormitoris:** De les habitacions que tenim, mostra el número d'aquelles que es consideren dormitoris

El número de **banys** de l'habitatge → Numèric

**Aire condicionat:** Indica si l'immoble està equipat o no amb l'aire condicionat → Binari

**Terrassa:** Indica si hi ha una terrassa o no. En el cas de que n'hi hagi, es posa el número de elles → Numèric. Per exemple '0' ('No'), 1,2,3,4,5. Els 'No' de la base de dades poden convertir-se a 0.

**Balcó:** el mateix que la columna de les terrasses → Numèric

**Sauna:** Indica si l'immoble està equipat amb una sauna → Binari

**Gimnàs:** Indica si l'immoble està equipat amb un gimnàs → Binari

**Piscina:** Indica si l'immoble està equipat amb una piscina o no. Si es privada, s'indica amb un número i si és comunitària, veurem la paraula "comunitària" → Literal

**Tipus de calefacció:** Bomba de calor, corrent elèctrica, gas → Literal

**Ascensor:** Indica si la propietat té ascensor → Binari

**Certificat d'eficiència energètica:** El certificat d'eficiència energètica o certificat energètic és un document oficial redactat per un tècnic competent que inclou informació objectiva sobre les característiques energètiques d'un immoble. → Binari. Pren els valors '0' i '1'. 'Sí' correspon a 1 i 'No' correspon a 0.

**Data de publicació de la propietat a la web:** Indica la data quan es va penjar la propietat a la web per primera vegada

**Venut:** Pren els valors '0' i '1'. 'Sí' correspon a 1 i 'No' correspon a 0. Binari.

I per últim, la columna de **"Data venda"** ens indica la data de quan es va vendre l'immoble. Si no hi ha cap data, vol dir que l'habitatge encara està en venda. → Literal. Data en què es ven.

---

#### 2.1.1. CAMPS NUMÈRICS: DISCRETS I CONTINUS

Els camps numèrics són valors quantitatius que es poden classificar en discrets i continus.

**Valors discrets:** Un nombre finit de valors entre dos valors qualssevol. El nostre cas, aquestes variables són per exemple: *número d'habitacions*.

**Valors continus:** Un nombre infinit de valors entre dos valors. Una variable contínua pot ser numèrica o de data / hora. En el nostre cas, aquests valors, en realitat són intervals numèrics com a per exemple: *l'any de construcció, superfície dels habitatges, preu d'habitatge*, etc

---

#### 2.1.2. ATRIBUTS QUALITATIUS

Aquests valors es refereixen a les qualitats específiques de les dades. Una qualitat és una propietat o un atribut que un objecte té. En el nostre cas aquests valors són: *Tipus de calefacció, districtes, barris*, etc.

### 2.2. TRACTAMENT DE DADES PREVI A L'ANÀLISI

El pas previ a qualsevol anàlisi de dades és efectuar un tractament per tal d'adequar els seus valors. Primer de tot hem d'aplicar un control de qualitat al fitxer "BBDD-Ampliada-Nova.xlsx", perquè pot haver-hi problemes d'ortografia i de majúscules/minúscules. També hi ha valors "null". D'altra banda, ens pot interessar convertir-lo a format CSV.

Per començar hem passat l'arxiu XLSX a CSV. Per desfer-me de les majúscules he optat per l'opció d'utilitzar (.lower).

Després s'han modificat les variables discretes o binàries (SI/NO) perquè només tinguin valors numèrics, així es podrà realitzar l'anàlisi de clúster K-means.

Hi ha variables que són numèriques però contínues com la superfície d'una propietat, el preu de l'habitatge i l'any de construcció. És per això que s'ha discretitzat el valor d'aquestes variables, utilitzant la funció *aprox* que apareix a l'arxiu **main.py**.

El codi d'aquesta funció és el que segueix:

```
def aprox(num, interval):  
    resta = num - (num // interval) * interval  
    if resta < (interval / 2.):  
        aproximacio = num - resta  
    else:  
        aproximacio = num + interval - resta  
    return aproximació
```

En el codi els anys de construcció de la propietat es discretitzaran de 10 en 10 anys.

`aprox(int(i[7]), 10)` on `i[7]` és el camp de l'any de construcció de la propietat.

El preu dels habitatges anirà de 200000 en 200000 euros.

`aprox(int(i[9]), 200000)` on `i[9]` és l'índex del camp del preu de les propietats.

La superfície de les propietats s'agruparà al voltant d'interval·ls de 50 en 50 m<sup>2</sup>.

`aprox(int(i[11]), 50)`, on `i[11]` és l'índex del camp de la superfície de les propietats.

### 2.3. K-MEANS AMPLIAT

A continuació es presenten els resultats del Kmeans en els districtes on hi ha dades significatives després d'observar com s'aglutinen les dades al voltant dels centroides.

## LES CORTS

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE
1.0	1967.9	0.4	478688.5	105.0	3.3	2.0	1.0	1.0	0.0	0.5	0.0	0.0	0.9	0.7 427
1.0	1968.1	0.4	1035312.5	229.0	5.7	3.7	1.0	2.3	0.7	0.5	0.0	0.0	0.9	0.7 640
0.3	1966.7	0.3	6266666.7	1250.0	6.3	5.7	1.0	3.0	0.0	3.0	1.0	1.0	1.0	1.0 3

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE
0.6	1957.6	0.4	709090.9	156.1	3.3	2.2	1.0	1.0	0.0	0.9	0.0	0.0	1.0	0.8 33
1.0	1967.4	0.3	1139130.4	252.2	5.7	3.8	1.0	1.9	0.7	1.0	0.0	0.0	1.0	0.8 23
0.0	1970.0	0.5	4450000.0	937.5	6.9	6.5	1.0	1.9	0.0	2.1	1.0	1.0	1.0	1.0 8

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE
0.9	1968.6	0.1	514285.7	100.0	2.9	1.6	0.9	0.7	0.3	0.9	0.0	0.0	1.0	0.4 7
0.5	1985.0	0.0	1600000.0	350.0	6.0	4.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0 2
0.3	1920.0	0.7	2266666.7	533.3	6.3	3.0	1.0	1.7	0.0	1.0	1.0	1.0	1.0	1.0 3

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE
0.7	1959.1	0.3	756410.3	167.9	3.6	2.2	1.0	0.9	0.1	0.9	0.0	0.0	1.0	0.8 78
0.8	1965.5	0.4	1127272.7	243.2	6.2	3.7	1.0	1.8	0.6	0.7	0.0	0.0	1.0	0.7 22
0.1	1963.9	0.3	3944444.4	825.0	7.0	4.6	1.0	1.6	0.0	1.8	1.0	0.9	1.0	1.0 18

### RESUM DADES GLOBALES DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	93.2 (640)	3.3 (23)	0.3 (2)	3.2 (22)
CENTROIDE 1	78.3 (427)	6.1 (33)	1.3 (7)	14.3 (78)
CENTROIDE 2	9.4 (3)	25.0 (8)	9.4 (3)	56.2 (18)

Figura 20 - Resultats del Kmeans de Les Corts

S'observa que les propietats que s'agrupen al voltant del centroide 1 són molt més grans que no pas les del centroide 0.

## Centroide 0

Els 640 habitatges al voltant del centroide 0 que s'han venut fàcilment disposen de més terrasses o balcons. Són en la seva totalitat pisos i són de mitjana 3 anys més noves que les que no s'han venut del mateix centroide. La resta de característiques és igual.

Les propietats que no s'han venut en aquest centroide algunes d'elles són cases (22) no tenen tanta sortida a l'exterior en forma de terrasses i balcons. El preu sembla proporcional a la superfície.

## Centroide 1

Les 427 propietats que s'agrupen al voltant del centroid 1 són pisos en la seva totalitat i la seva construcció va ser a finals dels anys 60.

Les 78 propietats que no s'han venut al voltant d'aquest centroid, són un 30% cases. Tenen al voltant d'un 50% més de superfície, però el seu preu és més car que aquest percentatge, per un nombre semblant de la resta de característiques i això que es van construir a finals dels 50.

Cal considerar si aquestes cases de les corts poden ser rebaixades de preu abans de vendre-les.

## CIUTAT VELLA

### MITJANES DE VIVENDES DISTRIBUÏDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFÍCIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1950.0	0.2	2240000.0	590.0	8.0	5.6	1.0	1.6	0.0	1.0	1.0	1.0	1.0	1.0	5
1.0	1963.2	0.3	473684.2	123.7	2.6	1.7	0.9	0.7	0.2	0.9	0.0	0.0	0.9	0.7	19
1.0	1970.0	0.4	1160000.0	310.0	5.0	3.6	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	5

### MITJANES DE VIVENDES DISTRIBUÏDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFÍCIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1987.5	0.0	2200000.0	600.0	5.0	5.8	1.0	1.8	0.0	1.0	1.0	1.0	1.0	1.0	4
1.0	1956.6	0.4	582965.3	154.7	3.0	1.3	1.0	0.2	0.1	0.8	0.0	0.0	1.0	0.9	317
1.0	1958.4	0.3	878095.2	231.9	5.7	3.6	1.0	2.5	0.6	0.9	0.0	0.0	1.0	0.9	105

### MITJANES DE VIVENDES DISTRIBUÏDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFÍCIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1970.0	0.5	2000000.0	500.0	4.5	3.0	1.0	1.0	0.0	1.0	1.0	0.5	1.0	1.0	2
1.0	1966.4	0.4	490909.1	127.3	3.5	2.1	0.9	0.9	0.1	1.0	0.0	0.0	1.0	0.7	11
1.0	1948.0	0.0	1080000.0	280.0	4.4	4.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	5

### MITJANES DE VIVENDES DISTRIBUÏDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFÍCIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1952.9	0.4	2171428.6	557.1	5.9	5.0	1.0	1.3	0.0	1.0	1.0	0.7	1.0	1.0	7
1.0	1962.9	0.4	579591.8	153.6	2.9	1.5	1.0	0.4	0.2	0.9	0.0	0.0	1.0	0.8	98
1.0	1957.8	0.4	1043243.2	275.7	5.3	3.7	1.0	1.8	0.3	0.8	0.0	0.0	1.0	0.9	37

### RESUM DADES GLOBALES DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	4.3 (19)	71.2 (317)	2.5 (11)	22.0 (98)
CENTROIDE 1	27.8 (5)	22.2 (4)	11.1 (2)	38.9 (7)
CENTROIDE 2	3.3 (5)	69.1 (105)	3.3 (5)	24.3 (37)

Figura 21 - Resultats del Kmeans de Ciutat Vella

La totalitat de les propietats de ciutat vella són pisos. En general les propietats que es situen al voltant del centroid 0 són més grans que no pas les que s'aglomeren al voltant del centroid 2.

### **Centroid 0**

Les que s'han venut de forma adequada al voltant d'aquest centroid tenen més terrasses i balcons i això sembla que interessa molt en aquest districte. A més les que es venen més són menys moblades que la resta. Les que es venen més també tenen de mitjana un nombre més gran d'habitacions tot i ser més petites.

Potser es podria ajustar el preu de les propietats amb més superfície per als clients que vulguin gaudir d'espai sense necessitar tantes habitacions.

### **Centroid 2**

En aquest cas, passa un fenomen normal però a la vegada interessant, ja que permet veure la potència del k-means. Hi ha 317 propietats que tot i ser més cares, més antigues i amb menys comoditats es venen molt millor que la resta. L'explicació pot ser que les propietats es trobin en un barri diferent del mateix districte.



## EIXAMPLE

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.9	1948.3	0.6	516666.7	122.9	3.6	2.1	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	24
0.3	1970.0	0.0	1133333.3	283.3	4.7	4.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	3
1.0	1900.0	1.0	2000000.0	450.0	8.0	6.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1956.8	0.4	531666.7	127.2	3.6	2.3	1.0	1.0	0.3	0.6	0.0	0.0	0.9	0.8	360
1.0	1956.3	0.4	1057142.9	249.7	6.1	4.0	1.0	2.9	1.0	0.5	0.0	0.0	0.9	0.8	196
0.0	1970.0	0.3	2266666.7	550.0	6.3	5.0	1.0	2.0	0.0	1.0	1.0	1.0	1.0	1.0	3

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1956.2	0.4	527464.8	124.4	3.6	2.3	1.0	1.0	0.3	0.6	0.0	0.0	0.9	0.7	568
1.0	1955.3	0.4	1057142.9	250.9	7.0	4.0	1.0	3.0	1.1	0.5	0.0	0.0	0.9	0.7	399

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1957.3	0.4	509144.5	122.0	3.5	2.2	1.0	0.9	0.3	0.7	0.0	0.0	0.9	0.7	339
1.0	1955.2	0.4	1081025.6	255.6	6.7	4.0	1.0	2.9	1.0	0.6	0.0	0.0	0.9	0.7	195
0.2	1964.0	0.0	2560000.0	600.0	7.0	5.4	1.0	1.6	0.0	1.2	1.0	1.0	1.0	1.0	5

### RESUM DADES GLOBS DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	0.4 (3)	24.7 (196)	50.3 (399)	24.6 (195)
CENTROIDE 1	1.9 (24)	27.9 (360)	44.0 (568)	26.3 (339)
CENTROIDE 2	11.1 (1)	33.3 (3)	0.0 (0)	55.6 (5)

Figura 22 - Resultats del Kmeans d'Eixample

En general, les propietats que s'acumulen al voltant del centroid 0 són molt més petites que les que s'acumulen al voltant del centroid 1. L'antiguitat de totes les propietats és molt semblant en tots els casos. Es parla de finals dels anys 50.

## Centroid 0

En aquest cas les propietats són molt semblants i no s'entén que unes es venguin adequadament altres amb dificultat i altres, senzillament no es venen. Una possible explicació és el barri dins del districte que ocupen cadascuna de les propietats. Cal valorar si s'han d'ajustar els preus de les propietats que no es venen o es venen amb dificultat o es pot afegir alguna característica que faci atractiva la seva adquisició.

## Centroide 1

Les propietats que es venen de forma adequada en aquest districte, tot i ser molt semblants en característiques a la resta de cases tenen un preu psicològic per sota del milió d'euros.

Hi ha 238 cases que no es venen que cal reconsiderar si estan ben valorades ja que tot i tenir menys comoditats que les 500 que es venen amb dificultat són més cares.

## GRÀCIA

MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.8	1957.9	0.5	410526.3	126.3	3.2	1.8	1.0	0.7	0.2	0.6	0.0	0.0	0.9	0.6	19
0.8	1957.5	0.2	950000.0	254.2	5.2	3.5	1.0	1.6	0.4	0.7	0.0	0.0	0.8	0.8	12
0.0	1930.0	0.7	2933333.3	833.3	5.7	6.3	1.0	2.0	0.0	1.0	1.0	1.0	1.0	1.0	3

MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1956.3	0.4	355000.0	101.0	3.0	2.0	1.0	0.5	0.5	0.6	0.0	0.0	1.0	0.7	200
1.0	1954.5	0.3	790747.3	222.2	5.1	3.6	1.0	2.0	0.6	0.6	0.0	0.0	1.0	0.8	281
0.1	1964.3	0.6	2371428.6	671.4	6.4	5.9	1.0	1.6	0.0	1.1	1.0	1.0	1.0	1.0	7

MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1965.3	0.4	348695.7	99.1	2.9	2.0	1.0	0.5	0.5	0.5	0.0	0.0	0.9	0.6	230
1.0	1963.1	0.4	801201.2	229.9	5.6	3.7	1.0	2.0	0.7	0.6	0.0	0.0	0.9	0.6	333

MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1962.7	0.4	357513.0	102.3	3.0	2.0	1.0	0.6	0.4	0.6	0.0	0.0	0.9	0.6	193
0.9	1963.1	0.4	863348.4	241.6	5.5	3.7	1.0	1.8	0.6	0.6	0.0	0.0	0.9	0.7	221
0.0	1968.0	0.6	2600000.0	740.0	7.2	5.2	1.0	1.6	0.0	1.4	1.0	0.8	1.0	1.0	5

RESUM DADES GLOBS DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	1.4 (12)	33.2 (281)	39.3 (333)	26.1 (221)
CENTROIDE 1	3.0 (19)	31.2 (200)	35.8 (230)	30.1 (193)
CENTROIDE 2	20.0 (3)	46.7 (7)	0.0 (0)	33.3 (5)

Figura 23 - Resultats del Kmeans de Gràcia

La majoria de propietats que s'agrupen al voltant d'aquests centroides són pisos, exceptuant un 10% de les propietats que s'acumulen al voltant del centroid 0 que són cases i no s'han venut.

Les propietats que s'acumulen al voltant del centroid 0 són molt més grans que les que s'acumulen al voltant del centroid 1.

### **Centroide 0**

Les propietats que es venen de forma adequada tot i ser més velles tenen totes ascensor i certificat d'eficiència energètica. S'assemblen força a les altres propietats que es venen amb dificultat o no es venen al districte, potser una explicació és la ubicació del barri dins del districte.

Les propietats que no es venen tenen menys característiques que les que es venen amb dificultat. Seria bo plantejar-se el preu i considerar els diferents casos, ja que el 10 per cent de les propietats que no es venen són cases dins del districte de Gràcia.

### **Centroide 1**

Les propietats que es venen de forma adequada són molt semblants a les que es venen amb dificultat o no es venen. Fins i tot, les que es venen millor són més cares. Una possible explicació és la ubicació de les propietats dins del districte de Gràcia.

## SANT ANDREU

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1968.4	0.3	554376.7	233.7	5.7	3.8	1.0	2.2	0.7	0.5	0.0	0.0	0.9	0.6	377
1.0	1969.0	0.4	272507.6	111.6	3.7	2.1	1.0	0.8	0.0	0.5	0.0	0.0	0.8	0.5	331
0.0	1980.0	0.0	2200000.0	900.0	10.0	8.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.4	1963.3	0.3	786666.7	326.7	5.5	3.7	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	15
0.9	1959.6	0.3	333333.3	138.9	3.3	1.9	0.8	0.6	0.3	0.7	0.0	0.0	0.9	0.4	27
0.0	1960.0	0.0	2000000.0	800.0	5.0	7.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.0	1938.0	0.6	840000.0	350.0	4.0	3.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	5
1.0	1985.8	0.3	266666.7	108.3	2.7	1.8	0.8	0.3	0.3	0.3	0.0	0.0	0.7	0.3	12

### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.3	1959.3	0.4	786666.7	330.0	5.5	3.3	1.0	1.2	0.1	0.9	0.0	0.0	1.0	1.0	30
1.0	1971.8	0.3	269879.5	112.0	3.1	1.9	0.7	0.4	0.4	0.5	0.0	0.0	0.8	0.3	83
0.0	1990.0	0.0	2400000.0	1000.0	6.0	6.0	1.0	2.0	0.0	1.0	1.0	1.0	1.0	1.0	1

### RESUM DADES GLOBALES DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	73.1 (331)	6.0 (27)	2.6 (12)	18.3 (83)
CENTROIDE 1	88.3 (377)	3.5 (15)	1.2 (5)	7.0 (30)
CENTROIDE 2	33.3 (1)	33.3 (1)	0.0 (0)	33.3 (1)

Figura 24 - Resultats del Kmeans de Sant Andreu

Les propietats al voltant del centroid 1 són molt més petites que les propietats al voltant del centroid 0.

### Centroid 0

Les propietats que es venen més són pisos. Les cases no es venen tant bé. Els pisos que es venen bé són més moderns que les cases (finals dels seixanta o finals dels cinquanta respectivament).

Tot i que són més petits els pisos que es venen millor tenen més habitacions i banys. La proporció preu-superfície fa que els pisos que es venen millor estan molt més proporcionats. Haurem de revisar els preus de les cases que es volen vendre a Sant Andreu, ja que potser d'entrada en aquest barri no hi ha tant d'interès per aquest tipus de propietats.

## Centroide 1

Les cases que més es venen amb facilitat tenen més habitacions més banys i més sortides a l'exterior i tot i que el preu és molt semblant es venen molt millor.

Potser s'hauria de considerar el preu de venda de les 84 cases que no s'han venut.

### SARRIÀ - SANT GERVASI

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.8	1960.9	0.3	652173.9	134.8	3.0	2.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	0.9	23
0.6	1952.0	0.0	1320000.0	270.0	5.8	4.0	1.0	1.6	0.6	0.8	0.0	0.0	1.0	1.0	5
0.0	1952.0	0.4	2600000.0	550.0	8.4	5.2	1.0	1.4	0.0	1.4	1.0	1.0	1.0	1.0	5

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1967.1	0.3	496703.3	105.5	3.0	2.0	1.0	0.6	0.4	0.6	0.0	0.0	1.0	0.8	273
1.0	1967.3	0.3	1105952.4	228.4	5.7	3.7	1.0	2.0	0.7	0.5	0.0	0.0	1.0	0.8	336
0.2	1956.0	0.3	2680000.0	560.0	6.1	4.0	1.0	1.7	0.0	1.3	1.0	0.9	1.0	1.0	10

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1965.5	0.4	533333.3	107.1	3.1	2.0	1.0	0.6	0.4	0.9	0.0	0.0	1.0	0.9	42
1.0	1967.9	0.3	1088372.1	225.6	5.7	3.7	1.0	2.0	0.7	0.9	0.0	0.0	1.0	0.9	43
0.0	1955.7	0.3	3142857.1	635.7	7.6	3.9	1.0	1.6	0.0	1.9	1.0	1.0	1.0	1.0	7

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.8	1960.4	0.4	613333.3	126.7	3.2	2.0	1.0	0.8	0.2	0.8	0.0	0.0	1.0	0.8	120
1.0	1970.4	0.4	1112195.1	229.9	5.7	3.8	1.0	1.9	0.7	0.5	0.0	0.0	1.0	0.8	82
0.1	1962.0	0.2	3313636.4	761.4	8.1	4.9	1.0	1.8	0.0	1.8	1.0	1.0	1.0	1.0	44

#### RESUM DADES GLOBALES DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	1.1 (5)	72.1 (336)	9.2 (43)	17.6 (82)
CENTROIDE 1	5.0 (23)	59.6 (273)	9.2 (42)	26.2 (120)
CENTROIDE 2	7.6 (5)	15.2 (10)	10.6 (7)	66.7 (44)

Figura 25 - Resultats del Kmeans de Sarrià - Sant Gervasi

La totalitat de les propietats que s'acumula al voltant del centroid 0 són pisos. Un 20% de les propietats que no es venen bé al voltant del centroid 2 són cases. Les propietats que s'acumulen al voltant del centroid 2 són molt més petites que les que s'acumulen al voltant del centroid 0.

## Centroide 0

Les propietats que es venen adequadament són molt semblants en tot, fins i tot en preu, a les que no es venen. Les que no es venen són 3 anys més noves que les que es venen adequadament. L'explicació pot ser la localització de les propietats en relació al districte de Sarrià-Sant Gervasi.

## Centroide 2

Les propietats que es venen millor tenen una millor rati de superfície, prestacions i preu. Hem de considerar el preu de les propietats que no es venen, algunes de les quals són cases. Més antigues que les propietats que es venen adequadament.

### SANTS-MONTJUÏC

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE FÀCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
1.0	1969.1	0.4	665551.8	231.1	5.7	3.7	1.0	2.3	0.7	0.5	0.0	0.0	1.0	0.8	299
1.0	1968.0	0.3	298924.7	101.1	3.5	2.0	1.0	1.0	0.0	0.6	0.0	0.0	1.0	0.8	186
0.0	1970.0	1.0	2333333.3	783.3	10.7	7.3	1.0	1.7	0.0	1.0	1.0	0.7	1.0	1.0	3

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE MITJÀ

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.3	1952.0	0.4	1160000.0	385.0	5.6	4.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	10
0.8	1974.5	0.2	354545.5	125.0	3.3	1.8	0.8	0.5	0.4	0.7	0.0	0.0	0.9	0.4	22
0.2	1974.0	0.2	2240000.0	780.0	7.4	7.0	1.0	1.4	0.0	1.0	1.0	1.0	1.0	1.0	5

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE DIFÍCIL

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.7	1963.3	0.3	866666.7	316.7	4.3	3.7	1.0	1.0	0.0	1.0	0.0	0.0	1.0	1.0	3
0.9	1970.9	0.3	254545.5	86.4	2.8	1.6	0.8	0.3	0.4	0.5	0.0	0.0	0.6	0.2	11
0.0	1935.0	0.0	1900000.0	650.0	7.5	7.0	1.0	1.0	0.0	1.0	1.0	0.5	1.0	1.0	2

#### MITJANES DE VIVENDES DISTRIBUIDES AL VOLTANT DELS CENTROIDS AMB CLASSE NO VENUT

TIPUS	ANY	MOBLAT	PREU	SUPERFICIE	HABITACIONS	BANYS	AIRE	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE	
0.3	1962.2	0.3	1021621.6	352.7	5.8	3.9	1.0	1.2	0.1	0.9	0.0	0.0	1.0	1.0	37
0.9	1973.0	0.3	344000.0	112.0	3.1	1.9	0.8	0.6	0.2	0.7	0.0	0.0	0.8	0.4	50
0.2	1965.0	0.2	2100000.0	737.5	8.5	5.8	1.0	1.2	0.0	1.0	1.0	1.0	1.0	1.0	4

#### RESUM DADES GLOBALES DE DADES DEL DISTRICTE AL VOLTANT DELS CENTROIDS

NÚMERO DE CENTROIDE	PERCENTATGE VENDA FÀCIL	PERCENTATGE VENDA ADEQUADA	PERCENTATGE VENDA DIFÍCIL	PERCENTATGE NO VENUTS
CENTROIDE 0	69.1 (186)	8.2 (22)	4.1 (11)	18.6 (50)
CENTROIDE 1	85.7 (299)	2.9 (10)	0.9 (3)	10.6 (37)
CENTROIDE 2	21.4 (3)	35.7 (5)	14.3 (2)	28.6 (4)

Figura 26 - Resultats del Kmeans de Sants – Montjuïc

### **Centroide 0**

Les propietats que es venen millor en aquest districte i al voltant d'aquest centroiden tenen millor característiques que les que no es venen i una millor proporcionalitat entre preu i superfície.

Hem de considerar la rebaixar del preu de les propietats que no es venen bé, sobretot les que estan per damunt de la mitjana aritmètica.

### **Centroide 1**

Les propietats que es venen millor són en la seva totalitat pisos. Els pisos que es venen millor tenen un número d'habitacions i de comoditats i més terrasses. És veritat que les que no es venen tenen molt més superfície. Però no tanta diferència com el preu indica. Després d'arribar al final dels comentaris del camins. Hem de plantejar si qualsevol comercial pot vendre una casa a la ciutat de Barcelona. Sembla la missió d'un comercial experimentat.

## 3. ESTUDI ESTADÍSTIC DE LES DADES

### 3.1. ATRIBUTS I PARÀMETRES ESTADÍSTICS INTERESSANTS

En el treball actual s'ha triat la mitjana aritmètica per presentar els resultats de cada districte. Aquesta mesura de centralització és la més característica. Per fer-ho, s'ha utilitzat el mòdul *statistics* i el mètode *statistics.mean* d'aquest.

Hi ha molts més paràmetres estadístiques fàcilment accessibles en aquest mòdul i que donen molta més informació de la dispersió de les dades com la desviació estàndard o paràmetres d'equilibri de la mostra com la mediana.

Els mètodes corresponents del mòdul *statistics* que es poden incloure en versions futures i útils d'aquest treball són els que presenta la figura inferior a la terminal de l'anaconda.

```
>>> import statistics
>>> dir(statistics)
['Decimal', 'Fraction', 'StatisticsError', '__all__', '__builtins__', '__cached_',
 '__doc__', '__file__', '__loader__', '__name__', '__package__', '__spec__',
 '_coerce', '_convert', '_counts', '_decimal_to_ratio', '_exact_ratio', '_isfinite',
 '_ss', '_sum', 'collections', 'groupby', 'math', 'mean', 'median', 'median_grouped',
 'median_high', 'median_low', 'mode', 'pstdev', 'pvariance', 'stdev', 'variance']
>>>
```

Figura 27 - Mòdul Statistics

Els més importants que es podrien afegir són *statistics.median* que correspon a la mediana i *statistics.stdev* que correspon a la desviació Standard.

Els atributs que s'han triat per realitzar els càlculs i gràfics estadístics són:

**Tipus:** Si es tracta de una propietat Pis o una Casa.

**Any:** Any de construcció de la propietat. Si apareix un any posterior a l'actual vol dir que l'habitatge està encara en construcció

**Moblat:** Si es ven moblat o sense mobles.

**Preu:** El que costa comprar la propietat.



**Superfície:** m<sup>2</sup> de l'habitatge.

**Habitacions:** Número d'habitacions de la casa.

**Aire:** si la casa disposa d'aire condicionat.

**Terrassa:** Si la propietat disposa de Terrassa.

**Balcó:** : Si la propietat disposa de balcó.

**Garatge:** Número de places de garatge que incorpora la propietat.

**Sauna:** Si l'habitatge disposa de sauna. Això és veritat en cases d'alt standing.

**Gimnàs:** Si l'habitatge disposa de gimnàs. Això és veritat en propietats d'alt standing.

**Ascensor:** Propietat amb ascensor o sense.

**CEE:** Si la propietat té el certificat d'eficiència energètica.

### 3.2. GRÀFICS DE LES DADES: HISTOGRAMES, ETCÉTERA

Perquè l'empresa pugui disposar de gràfics estadístics de les propietats desglossades per districte s'ha utilitzat *matplotlib*. Per poder extreure el diagrama d'un districte concret, s'ha utilitzat *Tkinter* que mostra una llista que, en clicar-hi, guarda la imatge del districte amb les mitjanes aritmètiques de les següents variables o camps de la base de dades: Any de construcció, preu en milers dels habitatges, superfície de les propietats, número d'habitacions, número de banys, número de places de garatge, tant per cent de propietats moblades, tant per cent de propietats amb aire condicionat, número de terrasses, tant per cent de propietats amb balcó, tant per cent de propietats amb sauna, tant per cent de propietats amb gimnàs, tant per cent de propietats amb ascensor, tant per cent de propietats amb certificat d'eficiència energètica (CEE).

Els gràfics obtinguts coincideixen amb els obtinguts posteriorment en l'apartat de Flask i Javascript.

Els gràfics es presenten a continuació i s'analitzen posteriorment en l'apartat següent:  
3.3 CONCLUSIONS DE LES DADES.

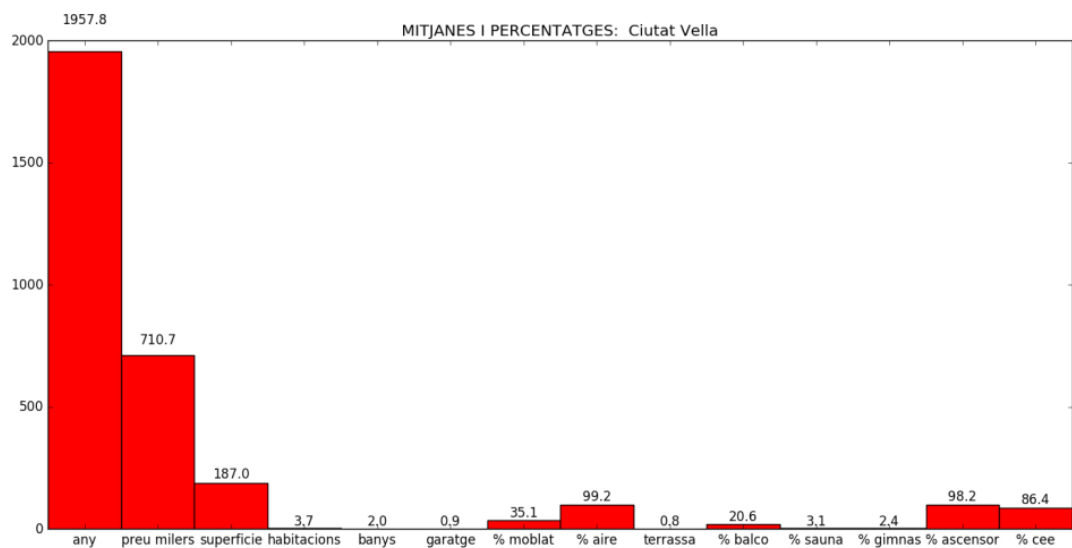


Figura 28 – Histograma – Ciutat Vella

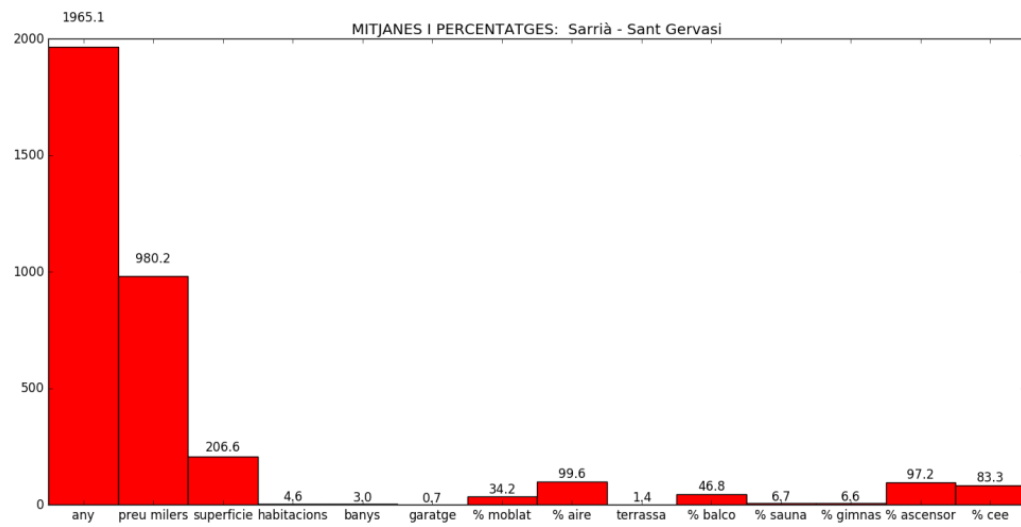


Figura 29 - Histograma - Sant Gervasi

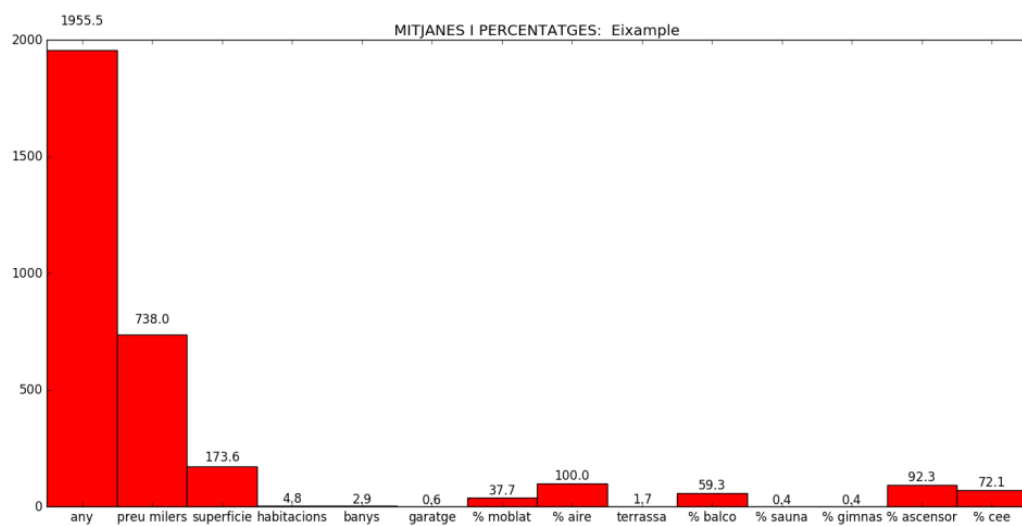


Figura 30 - Histograma - Eixample

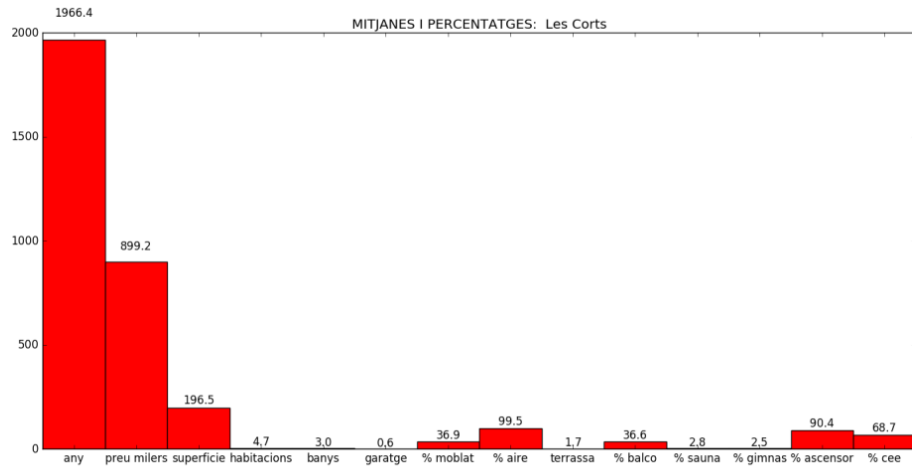


Figura 31 - Histograma - Les Corts

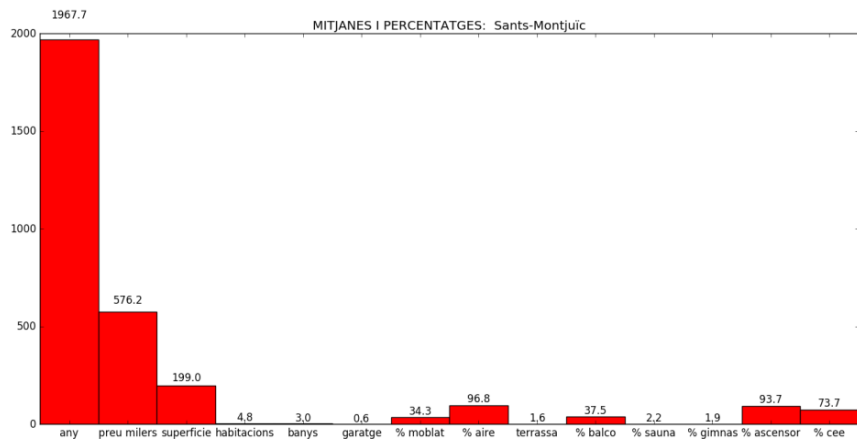


Figura 32 - Histograma - Sants Montjuïc

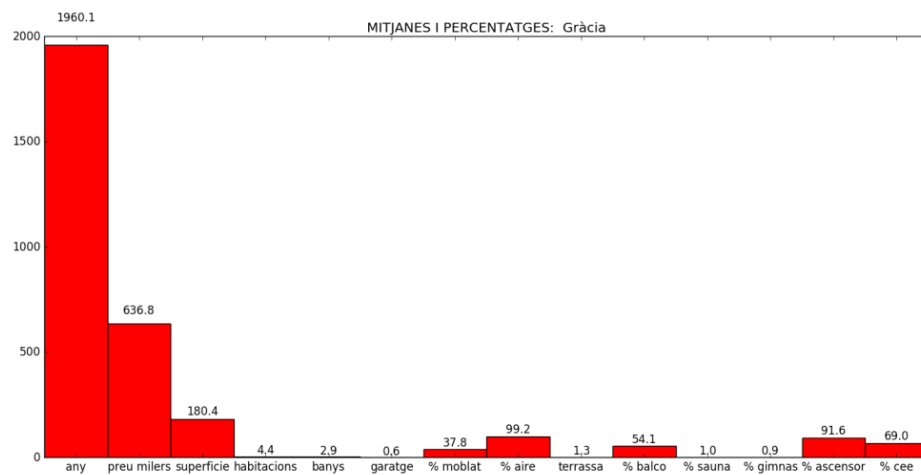


Figura 33 - Histograma - Gràcia

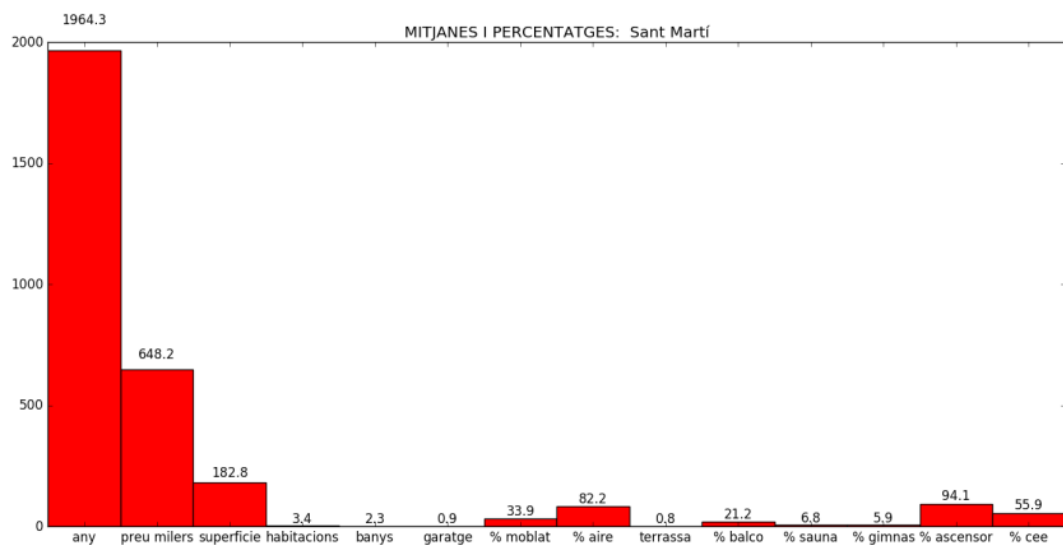


Figura 34 - Histograma - Sant Martí

### 3.3. CONCLUSIONS DE LES DADES

Aquest és el quadre resum de les estadístiques per districte de Barcelona . Hi ha alguns resultats molt evidents que entren dins del sentit comú i el coneixement previ i altres que són més sorprenents.

CARACTERÍSTICA	+	-
ANTIGUETAT	Eixample i Ciutat vella	Nou Barris
SUPERFÍCIE	Horta Guinardó i Sarrià sant Gervasi.	Sant Andreu
PREU	Sarrià Sant Gervasi i Les Corts	Nou Barris
HABITACIONS	Sants-Montjuïc, Eixample, Les Corts i Sant Andreu	Ciutat Vella, Nou Barris i Sant Martí.
BANYS	Sarrià-Sant Gervasi, Les Corts, Sants-Montjuïc	Ciutat Vella i Sant Martí
GARATGE	Sant Martí i Ciutat Vella	Nou Barris i Sant Andreu
MOBLAT	Horta-Guinardó	Sant Andreu
AIRE CONDICIONAT	Eixample i Sarrià-Sant Gervasi	Nou Barris.
TERRASSA	Eixample, Sarrià-Sant gervasi i les Corts	Nou Barris.
BALCÓ	Gràcia	Horta-Guinardó
SAUNA	Sarrià-Sant Gervasi	Eixample i Sant Andreu
GIMNÀS	Sarrià- Sant Gervasi	Sant Andreu
ASCENSOR	Ciutat Vella	Nou Barris
CEE	Sarrià-Sant Gervasi	Nou Barris

Algunes de les conclusions que es treuen són:

- **Sarrià-Sant Gervasi** té les propietats amb **més comoditats** i també **més cares**.
- **Nou Barris** té les propietats amb **menys comoditats** i també **més barates**.
- Les propietats amb **més ascensors** són les de **Ciutat Vella**.
- Les propietats amb **més balcons** són les de **Gràcia**.
- Les propietats amb **més garatges** són les de **Sant Martí** i **Ciutat Vella**.
- Les propietats **més antigues** són les de l'**Eixample** i **Ciutat Vella**.
- Les propietats **més grans** són les **d'Horta-Guinardó**.

- Les propietats **més moblades** són les d'**Horta-Guinardó**.
- Les propietats amb **més habitacions** són les de **Sants-Montjuïc, Eixample, Les Corts i Sant Andreu**.
- Les propietats amb **més terrasses** i aire condicionat són les de l'**Eixample** (tot i que **Sarrià-Sant Gervasi**, també gaudeix d'aquestes comoditats àmpliament).

## 4. CLASSIFICADOR DE LES DADES. ARBRES DE DECISIÓ

### 4.1. APLICACIÓ DE FILTRE PISOS VENUTS I NO VENUTS

D'entrada, per crear els arbres de decisió, no cal distingir entre pisos venuts i no venuts. Es defineix així un atribut que farà de classe que correspon a la facilitat de venda d'un pis.

### 4.2. DEFINICIÓ DE FACILITAT DE VENTA D'HABITATGE

Es considera que un pis es ven amb facilitat si es ven abans de 6 mesos. Si el pis es ven entre 6 mesos i 1 any a partir de la seva publicació a un portal de venda o immobiliària, es considera que és un interval mitjà o adequat de vendre. Si es triga més d'un any en vendre, es considera difícil de vendre.

### 4.3. CLASSES I ATRIBUTS

L'única classe que es farà servir per classificar els habitatges és la facilitat de venda. Els valors que pot prendre d'entrada aquest atribut classe són: "Fàcil", "Mitjà", "Difícil" i "No venut".

Els atributs que es fan servir per crear el diccionari que es carrega al mòdul *DTNum.py* són de propietat de l'*Andrés Cencerrado*.

Els atributs que s'han utilitzat són: Tipus, Any, Mobles, Preu, Superfície, Habitacions, Banys, Aire, Terrassa, Balcó, Garatge, Sauna, Gimnàs, Ascensor i CEE.

A la figura inferior es mostra el codi present a l'arxiu *main.py* amb el que es creen els arbres de decisió de cada districte de Barcelona.



```
llistadecisiontree=[]
for i in llistakmeans:
    if i[1]=='Les Corts' and i[-1]!='No venut\n':
        llistadecisiontree.append([i[0],i[3],i[4],i[5],i[6],i[7],i[8],i[9],i[10],i[11],i[12],i[13],i[14],i[15],i[16],i[len(i)-2],i[len(i)-1]])
for i in llistadecisiontree:
    i[-1]=i[-1].replace('\n','')
finallist=[]
for i in range(len(llistadecisiontree)):
    dict={}
    dict['Tipus']=float(llistadecisiontree[i][0])
    dict['Any']=float(llistadecisiontree[i][1])
    dict['Mobles']=float(llistadecisiontree[i][2])
    dict['Preu']=float(llistadecisiontree[i][3])
    dict['Superficie']=float(llistadecisiontree[i][4])
    dict['Habitacions']=float(llistadecisiontree[i][5])
    dict['Bany']=float(llistadecisiontree[i][6])
    dict['Aire']=float(llistadecisiontree[i][7])
    dict['Terrassa']=float(llistadecisiontree[i][8])
    dict['Balco']=float(llistadecisiontree[i][9])
    dict['Garatge']=float(llistadecisiontree[i][10])
    dict['Sauna']=float(llistadecisiontree[i][11])
    dict['Gimnas']=float(llistadecisiontree[i][12])
    dict['Ascensor']=float(llistadecisiontree[i][13])
    dict['cee']=float(llistadecisiontree[i][14])
    dict['class']=llistadecisiontree[i][16]
    finallist.append(dict)
```

Figura 35 - Arbres de decisió de cada districte

## 4.4. FUNCIO GOODNESS

La funció Goodness implementada és la que també es va desenvolupar i copiar al mòdul *DTNum.py* de l'Andrés Cencerrado que el consultor Samir Kanaan va proposar per tractar els arbres de decisió durant l'assignatura d'Intel·ligència Artificial Avançada.

Examinem el codi *DTnum.py* resulta que la funció *Goodness* que es troba al següent apartat (4.5).

## 4.5. ARBRE DE DECISIÓ

```
def goodness(classes, attr_values):
    best_cutoff = (0,0,'','')
    # (cutoff value, proportion of correctly classified instances, majority class up to cutoff point, majority class from cutoff point)
    resultat=[]
    # CALCUL DELS PUNTS DE TALL. PRIMER ELIMINEM ELS VALORS REPETITS. ELS PUNTS DE TALL ES TROBEN A LA MEITAT EXACTE
    # ENTRE DOS VALORS DE UN ATRIBUT

    llistaatributsordered=list(set(sorted(attr_values)))
    classeatributs=list(set(classes))
    puntsdetall=[]
    for i in range(len(llistaatributsordered)-1):
        puntsdetall.append((llistaatributsordered[i]+llistaatributsordered[i+1])/2.)
    # CALCULEM LA PROPORCIO DE CLASSE DE CADA PUNT DE TALL TAL COM SE EXPLICA A LA TEORIA.
    for punt in puntsdetall:
        comptadorsmenors=len(classeatributs)*[0]
        comptadorsmajors=len(classeatributs)*[0]
        for element in range(len(attr_values)):
            if attr_values[element]<punt:
                for k in range(len(classeatributs)):
                    if classes[element]==classeatributs[k]:
                        comptadorsmenors[k]=comptadorsmenors[k] + 1
            else:
                for k in range(len(classeatributs)):
                    if classes[element]==classeatributs[k]:
                        comptadorsmajors[k]=comptadorsmajors[k] + 1
        maxmenor=max(comptadorsmenors)
        maxmajor=max(comptadorsmajors)
        proporcio=(maxmenor+maxmajor)/float(len(attr_values))
        resultat.append((punt,proporcio,classeatributs[comptadorsmenors.index(maxmenor)],classeatributs[comptadorsmajors.index(maxmajor)]))
    resultatfinal=sorted(resultat, key=lambda x:x[1],reverse=True)
    # RETORNEM LA MILLOR TUPLA QUE RESULTA A SER EL PRIMER ELEMENT DESPRES DE ORDENAR ELS ELEMENTS PEL SEGON INDEX.
    best_cutoff=resultatfinal[0]
```

Figura 36 - Funció Goodness

Una vegada creada la llista de diccionaris d'un districte on cada diccionari representa un habitatge amb els seus atributs i classe, ja es crida el mòdul *DTnum* i en concret al seu mètode *DTnum.generateDT*.

S'ha de triar els nivells de profunditat del diccionari i l'exactitud mínima que se li demana al arbre de decisió.

Fins aquí tot és correcte, o això creiem. Els resultats han estat ben decebedors. Tal com es comenta en el següent apartat de conclusions.

## 4.6. CONCLUSIONS DE L'ARBRE DE DECISIÓ

Els resultats del classificador d'habitatges a partir dels arbres de decisió ha estat molt decebedors. Creiem que hi ha una gran disparitat en número d'habitatges a cada valor de la classe. La mostra d'elements de cada valor de classe són molt variables. Per exemple en un districte, la majoria d'habitatges (el 80 per cent) es venen amb facilitat i en canvi hi ha pocs individus a la resta de valors de classe.

En aquestes condicions el diccionari dona un error de codi *Out of Range* a la majoria de districtes excepte en els districtes on hi dos valors de classe molt clars. En aquest cas hi apareix un arbre de decisió molt senzill que només endevina els valors de classe dominants en un districte.

Per exemple, amb les proves fetes en l'execució del codi **main.py** amb un valor **max\_depth=10** i **min\_acuracy=0.8**. S'obté el següent arbre de decisió al districte de **Les Corts**:

```
[(Tipus,0.5,('Class','No venut'),('Class','fàcil'))]
```

El mateix arbre de decisió es troba al districte de **Sant Andreu**. Al districte de **Sants-Montjuïc** alternen dos arbres de decisió cada vegada que s'executa el codi de l'arxiu **main.py**.

```
['Preu',900000,('Class','fàcil'),('Class','No venut')]
```

o

```
[(Tipus,0.5,('Class','No venut'),('Class','fàcil'))]
```

Els altres districtes donen un missatge d'error de fora de rang amb aquests valors i altres de (**max\_depth** i **min\_acuracy**).

Si es treuen els habitatges "No venuts" i es vol tornar a repetir el procés de preparar les dades per a la creació d'un nou arbre de decisió els resultats segueixen sent igualment fèrtils, amb el coneixement que tenim d'arbres de decisió.

En aquest cas, amb els mateixos valors de **max\_depth=10** i **min\_acuracy=0.8**, s'obtenen els següents arbres de decisió (els districtes que no apareixen a la següent llista, donen un error de fora de rang, fins i tot provant diferents valors de **max\_depth** i **min\_acuracy**):

- **Les Corts:** [(('Tipus',0.5,('Class','Mitjà'),('Class','fàcil'))]
- **Sant Andreu:** [(('Any',1945),('Class',' Mitjà'),('Class','fàcil'))]

**[('Tipus',0.5,('Class','Mitjà'),('Class','fàcil'))]**

- **Sants-Montjuïc: idèntics resultats a Sant Andreu.**
- **Sarrià-Sant Gervasi: [('Habitacions',8.5),('Class','Mitjà'),('Class','Difícil')]**

**[('Aire',0.5),('Class','Difícil'),('Class',' Mitjà')]**

**[('Preu',3300000),('Class','Mitjà'),('Class','Difícil')]**

**[('Superfície',750),('Class','Mitjà'),('Class','Difícil')]**

**[('Garatge',1.5),('Class','Mitjà'),('Class','Difícil')]**

Aquests resultats són incomprensibles i només endevinen els valors de classe dominants segons l'anàlisi del kmeans que per sort si que ha donat resultats més evidents i fàcil d'interpretar.

Fins aquí tot és correcte, o això creiem. Els resultats han estat ben decebedors.

## 5. DADES DE CLIENTS DE LA NOSTRA BASE DE DADES D'HABITATGES

### 5.1. ANÀLISI DE LES DADES QUE ES TENEN : XLSX DE CLIENTS

---

#### 5.1.1. CAMPS NUMÈRICS: DISCRETS I CONTINUS

L'explicació dels camps numèrics discrets, continus i binaris, s'ha parlat àmpliament en els apartats següents de la present memòria: 2.1. i 2.2.

---

#### 5.1.2. ATRIBUTS QUALITATIUS

Els atributs qualitatius són els més problemàtics de tractar conjuntament amb dades numèriques. És molt interessant en un futur poder utilitzar l'algoritme K-means protocols que permeti calcular centroides amb atributs numèrics i categòrics barrejats.

Es pot consultar la bibliografia per poder entendre millor aquest punt que en una futura versió de l'aplicatiu serà plenament considerada.

[13] Implementation of the k-modes and k-prototypes clustering algorithms  
<https://pypi.python.org/pypi/kmodes/0.1> [Consulta novembre/desembre 2016]

## 5.2. TRACTAMENT DE DADES PREVI A L'ANÀLISI

Apart del tractament ja comentat a l'apartat 2.2, abans de l'anàlisi K-means, per exemple, cal normalitzar les dades perquè l'estudi sigui significatiu.

Al codi de l'arxiu *main.py*. Les dades es tracten de la següent manera abans de poder utilitzar-les a *k-means*:

- A) Es crea una array numpy de la llista que entreguem al k-means

```
matriu=np.array([i[0:len(i)-2] for i in llistacentral])
```

- B) Es normalitzen les dades perquè cap camp o atribut amb un valor absolut més gran tingui més pes que un altre:

```
normalizedarray=preprocessing.scale(matriu)
```

## 5.3. TIPUS DE CLIENTS. ANÀLISI K-MEANS CLUSTERING

K-means ha estat prou il·lustratiu del tipus de clients que hi ha a Barcelona en funció del districte i tipus d'habitatge. Les conclusions que s'han extret de forma general en el petit estudi de K-means desenvolupat a l'apartat 2.3.:

- Molts clients donen molta importància a terrasses i balcons.
- Els mobles del pis no interessen massa als clients que volen comprar una casa.
- Els clients són molt sensibles al districte i fins i tot al barri, tot i que d'entrada les ofertes que es troben no tinguin tantes comoditats o siguin més cares.
- Molts clients prefereixen més el número d'habitacions i de banys d'un pis que no la superfície total del pis o el tamany de les habitacions.

## 6. RECOMANACIÓ PELS NOSTRES CLIENTS

### 6.1. FILTRE DE PISOS NO VENUTS

La funció *full\_name(districte)* present a l'arxiu *main.py*, que s'enllaça amb la plantilla *seleccio.html*, és l'encarregada de filtrar les propietats no venudes de la base de dades utilitzada en aquest projecte.

Quan es carrega el web *flask* es troba el filtre *recommender* que es mostra a la figura inferior.

---

**PISOS RECOMANATS**

DISTRICTE	HABITACIONS	PREU	SUPERFICIE	MOBLES
Ciutat Vella ▾	0.0 ▾	200000 ▾	50 ▾	si ▾

[Torna a la pantalla de districtes](#)

Figura 37 - Pisos recomanats

L'enllaç corresponent és [127.0.0.1:5000/recomana](http://127.0.0.1:5000/recomana).

D'entrada es busquen les propietats del districte no venudes i que corresponen idènticament amb el districte, número d'habitacions, preu, superfície i si la propietat té mobles o no.

Aquest filtre es realitza dins de la plantilla *seleccio.html* amb l'ús del *Javascript* i cridant prèviament la funció *full\_name(districte)* definida en el *Python*. Aquesta crida ha estat un repte a nivell de programació.

## 6.2. FUNCIO SIMILITUD ENTRE HABITATGES

Quan no es troba una propietat idèntica a la cercada en la combinació de llistes desplegable (scroll-down lists) s'ideja un mètode que consisteix en els següents criteris:

- Primer es busquen propietats no venudes del districte que tinguin el preu més semblant tot i diferir de la resta de característiques
- En segon lloc, es busquen propietats que independentment del preu tinguin més similitud en la resta de característiques de les llistes desplegable (scroll-down list).

S'ha utilitzat una distància de Hamming adaptada al objectiu d'aquest treball que es recolza en la idea que dos camps iguals ponderen 1 i diferents no ponderen. Si dues propietats tenen dues característiques iguals això sumarà 2. Si les dues propietats tenen tres característiques iguals ponderen 3.

Primer, s'havia pensat en només afegir a les propietats recomanades les que complissin simultàniament en ambdós criteris (condició AND, simbolitzada &&). Finalment, donat que hi ha poques propietats s'ha optat per una condició que compleixi un dels dos criteris (condició OR, simbolitzada ||).

## 6.3. IMPLEMENTACIÓ D'UN RECOMMENDER

El criteri utilitzat en l'apartat 6.2 de la present memòria ha estat la base per crear el recommender. A continuació es mostren imatges que mostrin millor el procediment de programació i el resultat en el lloc web.



### PISOS RECOMANATS

DISTRICTE		HABITACIONS		PREU		SUPERFÍCIE		MOBLES	
Ciutat Vella		2.0		200000		50		sí	

TIPUS	DISTRICTE	BARRI	ANY	MOBLES	PREU	SUPERFÍCIE	HABITACIONS	BANYS	AIRE CONDICIONAT	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE
1.0	Ciutat Vella	La Barceloneta	1980.0	1.0	200000.0	50.0	2.0	2.0	1.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0

PER APROXIMACIÓ

TIPUS	DISTRICTE	BARRI	ANY	MOBLES	PREU	SUPERFÍCIE	HABITACIONS	BANYS	AIRE CONDICIONAT	TERRASSA	BALCO	GARATGE	SAUNA	GIMNAS	ASCENSOR	CEE
1.0	Ciutat Vella	El Raval	1970.0	1.0	200000.0	100.0	2.0	1.0	1.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0

[Torna a la pantalla de districtes](#)

Figura 38 – Recommender

En la figura superior es mostren dues propietats:

- A) La primera coincideix plenament en els criteris de la llista desplegable.
- B) La segona propietat es diferencia només en la superfície. Al no ser idèntica a la cerca del client es posa que és una recomanació PER APROXIMACIÓ.

Els camps que han de coincidir només són els de les llistes desplegables.

A continuació es mostra una imatge amb el codi que fa aquest filtre i que és la base del recommender. La programació s'ha fet en Javascript dins de la plantilla seleccio.html.

```
var maxim=Math.max.apply(Math,maxima);  
// FEM ELS ULTIMS PASSOS PER PRESENTAR ELS RESULTATS.  
indices=[];  
var cadenafinal='<table>';  
cadenafinal=cadenafinal + '<tr><th>TIPUS</th><th>DISTRICTE</th><th>BARRI</th><th>ANY</th><th>MOBLES</th><th>PREU</th><th>SUPERFÍCIE</th><th>HABITACIONS</th><th>BANYS</th><th>AIRE CONDICIONAT</th><th>TERRASSA</th><th>BALCO</th><th>GARATGE</th><th>SAUNA</th><th>GIMNAS</th><th>ASCENSOR</th><th>CEE</th></tr>';  
for(var i = 0, len = elements.length; i < len-1; i++) {  
    if (minima[i]==minim || maxima[i]==maxim){  
        indices[i]=i;  
    }  
}  
for(var i = 0, len = indices.length; i < len-1; i++) {  
    if (indices[i]!=undefined){  
        cadenafinal=cadenafinal + '<tr>';  
        for(var j = 0, len = elements[indices[i]].split(',').length; j < len; j++) {  
            cadenafinal=cadenafinal+'<th>' + elements[indices[i]].split(',') [j].replace(' ','') + '</th>';  
        }  
        cadenafinal=cadenafinal + '</tr>';  
    }  
}  
cadenafinal=cadenafinal+'</table>';
```

Figura 39 – Filtres de búsqueda de pisos

## 7. CREACIÓ D'UN LLOC WEB LOCAL

### 7.1. DJANGO VS FLASK



Figura 40 - Logo de Django - [Font](#)

**Django** és un marc de desenvolupament web de codi obert, escrit en Python, que respecta el patró de disseny conegut com a Model-vista-controlador. Va ser desenvolupat en origen per gestionar diverses pàgines orientades a notícies de la World Company de Lawrence, Kansas, i va ser alliberada al públic sota una llicència BSD al juliol de 2005; el framework va ser nomenat en al·lusió al guitarrista de jazz gitano Django Reinhardt.



Figura 41 - Logo de Flask - [Font](#)

**Flask** és un framework de desenvolupament micro web escrit en Python i basat en el conjunt d'eines i motor de plantilles Werkzeug Jinja2. És llicència de BSD.

No requereix eines o biblioteques particulars. No té cap capa de base de dades de l'abstracció, la validació de formularis, o qualsevol altre component a les biblioteques de tercers preexistents proporcionen funcions comuns.

Ens decantarem per Flask, perquè sembla més fàcil d'utilitzar i també per la recomanació del consultor.

Flask ha estat senzill d'utilitzar, en el cas d'aquest projecte s'han creat varis **decorators** que han donat un accés al web d'aquest treball.

A l'arxiu *main.py* després de tots els codis de càlcul del projecte de Python es crida als decorators que corresponen a l'esquema de la següent figura:

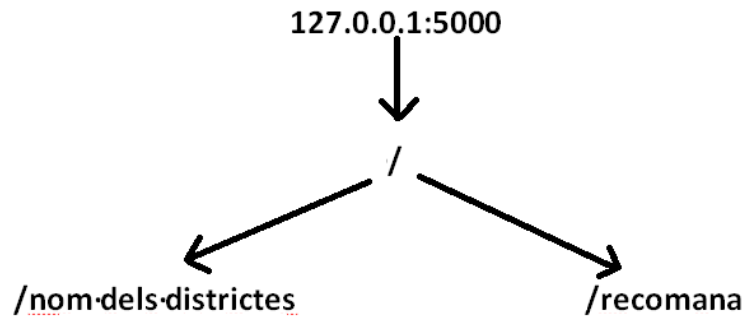


Figura 42 - Decorators

Cada decorator té una plantilla associada a la subcarpeta templates.

Quan executem *main.py* i s'obre automàticament el navegador per defecte a un ordinador en qüestió, en concret a l'adreça 127.0.0.1:5000, es carrega la plantilla *llistat.html*.

```
@app.route('/')
```

```
def index():
```

```
    return render_template("llistat.html", districtes=districtes, dectree=dectree)
```

El contingut de la plantilla *llistat.html* és el que es mostra a continuació:

```
<!doctype html>
</html>
<title>VIVENDES 1.0</title>
<head>
    <style>
```

```
body {  
    background-color: #F0E68C;  
}  
  
h4 {  
    text-align:center;  
}  
  
</style>  
</head>  
<body>  
<ul>  
    {% for item in districtes %}  
        <li><a href="/{{item}}">{{item}}</a></li>  
    {% endfor %}  
</ul>  
</body>  
</html>
```

La única cosa que es fa és cridar la llista de districtes del codi de Python i mostrar com enllaç html cadascun dels districtes. L'estil de la pàgina web es defineix a la mateixa plantilla *llista.html* amb la parella d'etiquetes `<style>` `</style>` en el head de l'arxiu html.

Els enllaços que mostra aquesta plantilla “obren la porta” al segon *decorator*, el contingut del qual es mostra a continuació.

```
@app.route('/<districte>')  
  
def barriada(districte):
```

```
return  
render_template("barry.html",districte=districte,dictcentral=dictcentral,district=distric  
t,camps=camps,diccionariskmeans=diccionariskmeans)
```

La plantilla *barry.html* és més complexa ja que les seves funcions són les següents:

- Mostrar la taula de mitjanes de les diferents variables de cada districte.
- Gràfic d'aquestes mitjanes per variable.
- Resum de dades globals de cada districte al voltant dels centroides del mètode k-means.
- Mitjanes de propietats distribuïdes al voltant dels centroides en funció de la classe de dificultat de venda de les propietats de les dades d'entrenament.

En aquesta plantilla s'utilitza *CSS* i *Javascript* al *head*, i *Html* al *body*.

Les anteriors plantilles *llistat.html* i *barry.html* corresponen a la part dels treballadors de la immobiliària. La última plantilla *seleccio.html* correspon a la part client accessible des de l'adreça 127.0.0.1:5000/recomana.

Aquesta plantilla es crida des del *decorator* de *main.py* que es mostra a continuació:

```
@app.route('/recomana')  
  
def recomana():  
  
    return render_template("seleccio.html",districtes=districtes,dictcentral=dictcentral)
```

Cal dir que aquesta plantilla ha estat la més difícil de realitzar ja que s'ha basat molt en Javascript amb els reptes que això comporta si es vol fer ús de llistes desplegables (dropdown lists) i tenir accés a les variables del Python utilitzant els mètodes `document.getElementById("nom de llista desplegable").value` i utilitzant una etiqueta en el body que s'actualitzi dinàmicament en funció dels valors de les llistes desplegables.

Per exemple, ha estat impossible accedir en abstracte a l'objecte `{{full_name(valordelallistade districtes)}}` del codi python i en comptes d'això ha calgut una astúcia de fixar-se en el valor que apareix a la llista desplegable de districtes i per ordre d'index entre dues llistes, seleccionar les propietats no venudes d'un districte en qüestió. Les llistes són les següents:

```
var item=['Ciutat Vella','Sarrià - Sant Gervasi','Eixample','Les Corts','Sants-Montjuïc','Gràcia','Horta - Guinardó','Nou Barris','Sant Andreu','Sant Martí'];
```

```
var filtredistricte=['{{full_name('Ciutat Vella')}}','{{full_name('Sarrià - Sant Gervasi')}}','{{full_name('Eixample')}}','{{full_name('Les Corts')}}','{{full_name('Sants-Montjuïc')}}','{{full_name('Gràcia')}}','{{full_name('Horta - Guinardó')}}','{{full_name('Nou Barris')}}','{{full_name('Sant Andreu')}}','{{full_name('Sant Martí')}}'];
```

Finalment, s'ha optat per fer la resta del codi d'aquesta plantilla amb Javascript i Html sense realitzar més crides al codi Python.

Ha resultat interessant veure que des de Flask no només es poden cridar llistes, diccionaris o altres objectes del Python, que seran accessibles utilitzant la gramàtica del doble bracket `{{ }}`, sinó també funcions Python utilitzant el mètode **`@app.context_processor`** . En concret, a les següents línies de codi de *main.py* es defineix la funció *full\_name(districte)* que és cridada des de la plantilla *jinja2, selecció.html*.

```
@app.context_processor
def some_processor():
    def full_name(districte):
        caca=""
        for i in llistakmeans:
            if districte in i and 'No venut\n' in i :
```

```
cucu=cucu+str(i[0:-2])  
  
return cucu  
  
return {'full_name': full_name}
```

A nivell de Javascript s'han comprovat les següents particularitats d'aquest llenguatge script si el comparem amb el funcionament del Python:

- Les llistes importades des del Python es tracten com *strings* i es per això que després cal “jugar” amb els mètodes Split i replace per aconseguir l'accés individuals als camps de la base de dades de propietats.
- Les *arrays* (llistes) Javascript funcionen diferent ja que si s'inicialitza una nova array dins d'un bucle for utilitzant un índex que varia segons la longitud d'una altra *array* al no haver-hi el mètode *append* del Python et crea una altra *array* amb la mateixa longitud però amb molts camps *undefined*. L'array queda definida com es mostra a continuació, per exemple:

```
[,,,,,,14,,,23,,,,38,,,,60,,,,,,,,]
```

On els *index undefined*, que no tenen cap valor, s'han de tractar posteriorment com *undefined*.

- Dins de les condicions and s'utilitza com && i or com || i cal distingir entre els comparadors == i ===. On === és un tipus de comparació estricta on apart del valor es consulta la igualtat del tipus. Cal tenir ben present aquest aspecte al utilitzar-lo en comparacions condicionals if.
- Per assegurar que es podran realitzar operacions matemàtiques entre variables es pot utilitzar l'operand *parseInt()* i així es pot assegurar que es treballa amb camps numèrics.
- Per trobar el valor màxim o mínim d'un array , que s'ha utilitzat en el recommender dels clients, cal utilitzar Math.max.apply(Math,nom de l'array) i Math.min.apply(Math,nom de l'array) respectivament.

- La seva relació amb l'html fa que sigui molt interessant afegir a les stri()ngs tot el que es vulgui presentar en el body. Per exemple str=cadena + '</br>' + '<h4>NOVA LLISTA</h4>' + cadena2.
- Per poder presentar els resultats del javascript en el body es pot utilitzar una etiqueta en el body ( per exemple, <p id="demo"></p>) que s'actualitza amb el mètode:  
`document.getElementById("demo").innerHTML`  
on demo és l'id de l'element html.
- El Javascript s'afegeix amb el parell d'etiquetes <script></script> ja sigui en el head o en el body del document html.  
Cal vigilar l'ús del comandament document.write() ja que moltes vegades interfereix amb la resta de codi. Per comprovar o "printar" el resultat de certes operacions amb objectes javascript millor utilitzar alert() o modificar "temporalment" el valor de document.getElementById("demo").innerHTML
- Les línies de codi Javascript finalitzen amb punt i coma (;) i això a vegades és una font de problemes per les persones que venen del python, per exemple.
- Els bucles for o els condicionals if utilitzen els brackets {} en comptes dels dos punts. Cal utilitzar la regla de tants oberts, tants tancats en tot moment.



## 7.2. TARGET DEL LLOC WEB

El target de la nostra pàgina web és per als treballadors de la immobiliària i també algunes parts per als nostres clients.

Tal com s'ha explicat en l'apartat 7.1 Django vs Flask el mateix arxiu *main.py* utilitza decoradors per la part client i la part empresa.

Cada vegada que s'executa Python *main.py* es tornen a calcular totes les variables i s'executa el codi python en la seva totalitat.

Per separar el codi de la part servidor i client caldria realitzar els càlculs amb Python per emmagatzemar en un arxiu independent les variables com: *llistadellistestotal*, *llistakmeans*, *districtes*, *llistacentral*, *centroides*, *centroides*, *llistadecisiontree*, *dectree* i altres.

La idea és executar tot el codi python i després separar els decoradors i plantilles de la part de l'empresa i de la part client. L'esquema és el següent:

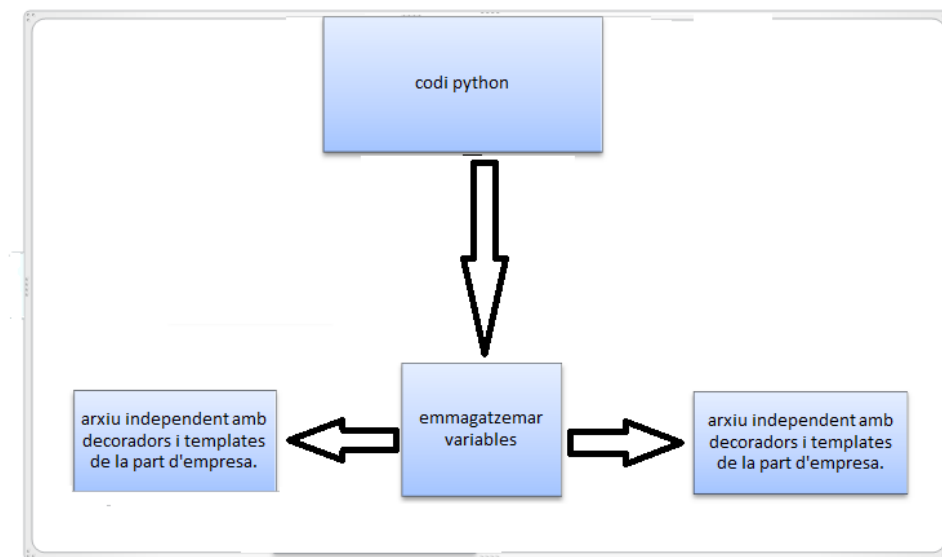


Figura 43 – Esquema de la web

De moment, el codi actual s'executa en un sol arxiu i calculant les variables, centroides del k-means, llistes, diccionaris cada vegada, però això es pot millorar i fer que el servidor flask s'executi àgilment amb les variables calculades prèviament, així

s'aconseguirà un projecte independent al màxim dels càlculs Python. La forma de fer-ho és calcular les diferents llistes i redirigir els resultats a un arxiu que convenientment manipulat serà la base per definir les variables calculades. En un nou arxiu *main.py* s'utilitzaran les variables amb els valors calculats prèviament calculats i llavors s'arrencarà el flask de la part client o de la part d'empresa segons calgui.

### 7.3. ESQUEMA DEL LLOC WEB I USABILITAT

El lloc web serà compatible amb els 4 navegadors importants: Google Chrome, Firefox, Safari i Edge. També serà *responsive*, el que vol dir que serà perfectament visible i compatible als dispositius mòbils i tablets. En aquest projecte només es farà la versió *desktop*.

### 7.4. INTERELACIÓ DEL LLOC WEB AMB CODI PYTHON IMPLEMENTAT

Quan tinguem l'aplicació al codi de Python, hem servit el Framework de Flask perquè era molt més senzill per a un usuari obrir l'aplicació des d'una web que executant el codi.

### 7.5. INTERPRETACIÓ DEL LLOC WEB

A continuació es detallarà l'accés al lloc web i el seu funcionament per interpretar correctament el lloc web.

## 8. ADAPTACIÓ DEL LLOC WEB PER UTILITZAR-LO A INTERNET

### 8.1. INTEGRACIÓ A DIFERENTS NAVEGADORS

La versió d'aquesta eina és compatible amb la majoria dels navegadors del mercat. En concret, s'ha provat amb èxit amb els següents navegadors per a ordinador portàtil o sobretaula amb els següents navegadors.

- Microsoft Internet Explorer Versió 11
- Mozilla Firefox Version 30 i posterior 50.1.0
- Google Chrome versió 55.0.2883.87
- Safari (totes les versions) 5.1.7
- Opera Stable 42.0.2393.94



Figura 44 – Navegadors – [Font](#)

## 9. MILLORES I FUTURES ACTUALITZACIONS

Aquest projecte és objecte de millora en un futur immediat. De fet, pot ser interessant en el món de les immobiliàries. Els punts que cal reforçar són:

- Es necessita una base de dades on els camps de data de publicació d'una propietat i data de venda sigui real. Aquestes dades no són accessibles fàcilment ni des de llocs web ni des d'empreses. Això és bàsic per poder utilitzar les etiquetes de classe de dificultat de venda: 'fàcil' 'mitjà' 'difícil' i 'No venut'. A partir d'aquí es poden entrenar correctament els algoritmes k-means o arbres de decisió amb més possibilitat d'èxit de tenir un bon classificador. Aquest classificador seria molt útil per poder donar els pisos de venda difícil a un tipus de comercial o un altre.
- Cal experimentar amb l'ús del servidor web amb diferents plataformes i el seu ús a xarxes d'àrea local i Internet. Així mateix, cal provar la seva exportabilitat a altres plataformes com smartphones (IOS o Android) i tablets.
- Cal millorar l'estètica de la pàgina web, per fer-la atractiva. El client ha de poder visualitzar les propietats que apareixen en les seves cerques, ja siguin les propietats que coincideixen exactament amb les opcions del client, com les propietats recomanades que més s'acosten a les seves necessitats.
- Possibilitat de validació d'usuaris i login per poder enviar ofertes al clients de la base de dades.
- Adaptació del projecte perquè sigui multilingüe.

Més endavant es farà una app per a versió Android i una altra per versió de Apple. L'app podrà ser trobada tant a Play Store com a iTunes i serà gratuïta.

## 10. PRESSUPOST DEL PROJECTE

Pressupost			
Nom: Goran Polic		Dades client	
Adreça:		Immobiliària Barcelona	
Població: Barcelona		Adreça: Diagonal 640	
Província: Barcelona		Població: Barcelona	
CIF / NIF: 999999999		Província: Barcelona	
Telèfon/fax: 93 99 999 99		CIF / NIF: 111111111	
Correu electrònic:			
Data del pressupost	27-08-27	<u>Validesa: 01/11/2016</u>	-
Mes	Descripció	TOTAL	
Octubre	Anàlisi de les dades	500,00 €	
Octubre	Estudi estadístic de les dades	500,00 €	
Octubre	Programació	500,00 €	
Novembre	Programació	2.500,00 €	
Desembre	Programació	2.500,00 €	
Desembre	Creació d'un lloc web local	1.000,00 €	
Desembre	Adaptació del lloc web per utilitzar-lo a internet	500,00 €	
Desembre	Redacció del projecte	500,00 €	
Desembre	Manual d'usuari	500,00 €	
		9.000,00 €	
	IVA: 18%	10.620,00 €	
Total pressupost			10.911,60 €
Forma de pagament :			
		Accepto el pressupost	

## 11. CONCLUSIONS

En aquest projecte la part molt importat és la base de dades que farem servir. Segons la qualitat i la quantitat de la mateixa, obtindrem uns resultats o altres. En el nostre cas, hem obtingut un arxiu de 8416 entrades que havíem de preparar per poder treballar: preparar els camps necessaris, destriar aquells que no necessitem. També discretitzar valor continuus que ens ha suggerit el consultor: crear intervals amb valors com la superfície, preu, etc.

El projecte s'ha centrat en 3 punts:

- Estadístiques, que és un pas previ en Intel·ligència Artificial que et dona una certa idea sobre les propietats: aquelles amb més comoditats i també més cares, amb menys comoditats i també més barates, els districtes amb més ascensors, balcons, garatges, habitatges més antics, etc
- El segon punt important són els Kmeans on hi ha dades significatives després d'observar com s'aglutinen les dades al voltant dels centroides. Les propietats que es venen millor són en la seva totalitat pisos. Els pisos que es venen millor tenen un número d'habitacions i de comoditats i més terrasses.
- Arbres de decisió: Els resultats del classificador d'habitatges a partir dels arbres de decisió ha estat molt decebedor. Creiem que hi ha gran disparitat en número d'habitatges a cada valor de la classe. La mostra d'elements de cada valor de classe són molt variables. Per exemple en un districte, la majoria d'habitatges (el 80 per cent) es venen amb facilitat i en canvi hi ha pocs individus a la resta de valors de classe.

El propòsit final d'aquest projecte ha estat el fet de poder fer servir el Python, TKinter, Flask i Javascript per obtenir resultats gràfics que són més "user friendly" per a l'usuari final.

Amb una base de dades més gran, haguéssim obtingut uns resultats més complets.

## 12. GLOSSARI

<b><i>Taxació</i></b>	És un document legal que certifica el valor de l'immoble per a una finalitat descrita, realitzat amb una metodologia basada en una normativa legal.
<b><i>Valoració</i></b>	És un informe que estableix el valor orientatiu d'un immoble mitjançant l'aplicació de diversos mètodes de valoració, però que no té una validesa legal, en no ajustar-se a un procediment valoratiu normat.
<b><i>CEE</i></b>	Certificat eficiència energètica: És un document oficial redactat per un tècnic competent que inclou informació objectiva sobre les característiques energètiques d'un immoble.
<b><i>Intel·ligència artificial</i></b>	La intel·ligència artificial, com a camp de la informàtica que és, està dedicada a la construcció de programes.
<b><i>Lead</i></b>	És l'acció quan una persona / usuari ens ha facilitat les seves dades de contacte mitjançant el formulari d'una web, aplicació mòbil, etc

## 13. BIBLIOGRAFIA

[1] Districtes i barris de Barcelona

[https://es.wikipedia.org/wiki/Distritos\\_de\\_Barcelona](https://es.wikipedia.org/wiki/Distritos_de_Barcelona) [Consulta setembre 2016]

[2] Las tasadoras pisan el freno: algunas valoran por debajo del precio de compra dificultando la concesión de hipotecas

<https://www.idealista.com/news/finanzas/hipotecas/2016/09/20/743603-las-tasadoras-pisan-el-freno-valoran-por-debajo-del-precio-de-compra-dificultando-la>  
[Consulta setembre 2016]

[3] Evolución precios pisos metro cuadrado en venta en Barcelona

<http://www.habitacalia.com/informes/precio-medio-viviendas-barcelona.htm>

[Consulta setembre 2016]

[3] Informe del precio de la vivienda en venta

[http://www.fotocasa.es/indice-inmobiliario\\_fotocasa.aspx](http://www.fotocasa.es/indice-inmobiliario_fotocasa.aspx) [Consulta setembre 2016]

[4] Precios de viviendas en venta en Barcelona

<http://www.nuroa.es/venta/precios-de-vivienda-en-barcelona>

[Consulta setembre 2016]

[5] ¿Cuánto vale mi casa de Barcelona? <http://blog.cliventa.com/cuanto-vale-mi-casa-de-barcelona> [Consulta setembre 2016]

[6] Valoraciones viviendas: <http://www.valoracion.es/valoraciones-de-viviendas/que-diferencias-existen-entre-tasacion-y-valoracion> [Consulta setembre 2016]

<https://librosweb.es/libro/python> [Consulta maig 2016]

[7] Apunts de l'assignatura Intel·ligència Artificial Avançada (UOC)



[Consulta setembre 2016]

[8] <http://www.learnpython.org/es> [Consulta setembre 2016]

[9] Anàlisi de Components Principals <https://vimeo.com/109796748>

[Consulta setembre 2016]

[10] Django (framework) [https://es.wikipedia.org/wiki/Django\\_\(framework\)](https://es.wikipedia.org/wiki/Django_(framework)) [Consulta novembre 2016]

[11] Flask (web framework) [https://en.wikipedia.org/wiki/Flask\\_\(web\\_framework\)](https://en.wikipedia.org/wiki/Flask_(web_framework))  
[Consulta novembre 2016]

[12] Automated Variable Weighting in k -Means Type Clustering <https://goo.gl/ZfE4r8>  
[Consulta novembre 2016]

[13] Implementation of the k-modes and k-prototypes clustering algorithms  
<https://pypi.python.org/pypi/kmodes/0.1> [Consulta novembre 2016]

## 14. ANNEXOS

### 14.1. CODI FONT ARXIU MAIN.PY

```
import io
import matplotlib.pyplot as plt
import numpy as np
from sklearn import preprocessing
from sklearn.cluster import KMeans
from tkinter import *
from flask import Flask, request, render_template
import webbrowser
import numpy as np
import statistics
import Dtnum
# FUNCIO APROX QUE SERVEIX PER DISCRETITZAR ELS VALORS DE SUPERFICIE,
PREU DE LES VIVENDES.
def aprox(num, interval):
    resta=num-(num//interval)*interval
    if resta < (interval/2.):
        aproximacio=num-resta
    else:
        aproximacio=num+interval-resta
    return aproximacio
#BASE DE DADES AMPLIADA QUE ES FA SERVIR PER REALITZAR L'ESTUDI DE
VIVENDES, CONSTA DE MES
DE 8000 VIVENDES.
fitxer=io.open('BBDD-Ampliada-Nova.csv','r')
llistastring=fitxer.readlines()
llistadellistestotal=[]
llistakmeans=[]
for i in llistastring:
    llistadellistestotal.append(i.split(';'))
for i in llistadellistestotal[1:]:
    #TRACTAMENT DE CAMPS
    if len(i[30]) != 1:
        if i[1]=='Pis':
            i[1]='1'
        else:
            i[1]='0'
        if i[8]=='Moblat':
            i[8]='1'
        else:
            i[8]='0'
        i[9]=i[9].replace('.', '')
        if i[16]=='No':
            i[16]='0'
        else:
            i[16]='1'
        i[17]=i[17].replace('No', '0')
        i[18]=i[18].replace('No', '0')

        i[19]=i[19].replace('No', '0')
        if i[24]=='No':
            i[24]='0'
        else:
            i[24]='1'
        if i[25]=='No':
            i[25]='0'
        else:
```

```
i[25]='1'
l·listakmeans.append([i[1],i[4],i[5],aprox(int(i[7]),10),i[8],aprox(int
(i[9]),
200000),aprox(int(i[11]),50),i[13],i[15],i[16],i[17],i[18],i[19],i[20]
,i[21],i[24
],i[25],i[29],i[30]])
#UNA VEGADA TRACTAR CERTS CAMPS DE LA BASE DE DADES JA ES PODEN
CONVERTIR EN FLOAT PREVI PAS
A INT DE ALGUN CAMP STRING. AIXI ES PODRAN TRACTAR EN DIFERENTS
METODES
#I PROCEDIMENTS.
for i in range(len(l·listakmeans)):
    for j in range(len(l·listakmeans[0])):
        if j!=1 and j!=2 and j!=len(l·listakmeans[0])-1 and
j!=len(l·listakmeans[0])-2:
            l·listakmeans[i][j]=float(l·listakmeans[i][j])
            l·listakmeans=l·listakmeans[1:len(l·listakmeans)]
#ES CREA UNA L·LISTA AMB TOTS ELS DISTRICTES
districtes=[]
for i in l·listakmeans:
    if i[1] not in districtes:
        districtes.append(i[1])
#####
#####
#####
#A PARTIR DE ARA TREBALLAREM AMB l·listakmeans QUE ES LA BASE DEL K-
MEANS TAMBÉ UTILITZADA A
LA PART DE CLIENT DE FLASK
#####
#####
#####
# district ES UN DICCIONARI ON EMMAGATZAMEM LES MITJANES DELS
DIFERENTS CAMPS DEL DISTRICTE.
# dictcentral ES UN DICCIONARI QUE EMMAGATZEMA INFORMACIO SOBRE DADES
ESTADISTIQUES DE CADA
CENTROID.
# l·listacentral ES UNA L·LISTA QUE UTILITZEM PER REALITZAR ELS CALCULS
K-MEANS A PARTIR DE LA
L·LISTA l·listakmeans.
# AQUESTES VARIABLES LES DEFINIM AQUI ABANS DE ENTRAR AL BUCLE QUE
ITERARA AL VOLTANT DELS
DIFERENTS DISTRICTES
dictcentral={}
district={}
postanalisi=[]

#CODI ESTADISTIQUES EN UN BUCLE QUE ITERA EN TOTS ELS DISTRICTES DE
FORMA INDIVIDUAL.

for districte in districtes:
    any=[]
    preu=[]
    superficie=[]
    habitacions=[]
    banys=[]
    garatge=[]
    tipus=[]
    mobles=[]
    aire=[]
    terrassa=[]
    balco=[]
    sauna=[]
    gimnas=[]
    ascensor=[]
    cee=[]
    subtipus=[]
    estat=[]
```

```
piscina=[]
calefaccio=[]
camps=['any','preu
milers','superficie','habitacions','banys','garatge','%
moblat','% aire','terrassa','% balco','% sauna','% gimnas','%
ascensor','% cee']
valors=[]
for i in llistadellistestotal[1:]:
if districte in i and len(i[7])!=0:
subtipus.append(i[2])
estat.append(i[3])
piscina.append(i[22])
calefaccio.append(i[23])
any.append(int(i[7]))
preu.append(int(i[9].replace('.','')))
superficie.append(int(i[11]))
habitacions.append(int(i[13]))
banys.append(int(i[15]))
garatge.append(int(i[19].replace('No','0')))
terrassa.append(int(i[17].replace('No','0')))
balco.append(int(i[18].replace('No','0')))
tipus.append(int(i[1]))
mobles.append(int(i[8]))
aire.append(int(i[16]))
sauna.append(int(i[20]))
gimnas.append(int(i[21]))
ascensor.append(int(i[24]))
cee.append(int(i[25]))
if len(any)!=0 and len(preu)!=0 and len(superficie)!=0 and
len(habitacions)!=0:
valors.append(statistics.mean(any))
valors.append(statistics.mean(preu)/1000)
valors.append(statistics.mean(superficie))
valors.append(statistics.mean(habitacions))
valors.append(statistics.mean(banys))
valors.append(statistics.mean(garatge))
valors.append(100*statistics.mean(mobles))
valors.append(100*statistics.mean(aire))

valors.append(statistics.mean(terrassa))
valors.append(100*statistics.mean(balco))
valors.append(100*statistics.mean(sauna))
valors.append(100*statistics.mean(gimnas))
valors.append(100*statistics.mean(ascensor))
valors.append(100*statistics.mean(cee))
district[districte]=valors
#####
#####
#####CODI DEL
KMEANS#####
#####
#####
#####
#####
#####
llistacentral=[]
for i in llistakmeans:
if districte in i:
llistacentral.append([i[0],i[3],i[4],i[5],i[6],i[7],i[8],i[9],i[10],i[
11
],i[12],i[13],i[14],i[15],i[16],i[len(i)-2],i[len(i)-1]])
if len(llistacentral)<4:
pass
else:
matriu=np.array([i[0:len(i)-2] for i in llistacentral])
```

```
normalizedarray=preprocessing.scale(matriu)
idx = KMeans(init='k-means++', n_clusters=3, n_init=10)
idx.fit(normalizedarray,y=None)
centroids=idx.cluster_centers_
centroids=idx.fit_predict(normalizedarray,y=None)
centroidistrictes=[]
for j in range(4):
    try:
        easy=0
        medium=0
        difficult=0
        notsolded=0
        for i in range(len(centroids)):
            if centroids[i]==j:
                if llistacentral[i][len(llistacentral[0])-1]=='Fàcil\n':
                    easy=easy+1
                postanalisi.append([districte,j,'Fàcil',llistacentral[i]])
                elif llistacentral[i][len(llistacentral[0])-1]=='Mitjà\n':
                    medium=medium+1
                postanalisi.append([districte,j,'Mitjà',llistacentral[i]])
                elif llistacentral[i][len(llistacentral[0])-1]=='Difícil\n':
                    difficult=difficult+1
                postanalisi.append([districte,j,'Difícil',llistacentral[i]])
            else:
                notsolded=notsolded+1
            postanalisi.append([districte,j,'No venut',llistacentral[i]])
        centroidistrictes.append(str(round(easy*100/(easy+medium+difficult+notsolded),1))+str(' (')+str(easy)+str(') ')))
        centroidistrictes.append(str(round(medium*100/(easy+medium+difficult+notsolded),1))+str(' (')+str(medium)+str(') ')))
        centroidistrictes.append(str(round(difficult*100/(easy+medium+difficult+notsolded),1))+str(' (')+str(difficult)+str(') ')))
        centroidistrictes.append(str(round(notsolded*100/(easy+medium+difficult+notsolded),1))+str(' (')+str(notsolded)+str(') ')))
    except:
        pass
dictcentral[districte]=centroidistrictes
#####
#####
#####
##### ARBRES DE DECISIO
#####
#####
#####
llistadecisiontree=[]
for i in llistakmeans:
    if i[1]=='Les Corts' and i[-1]!='No venut\n':
        llistadecisiontree.append([i[0],i[3],i[4],i[5],i[6],i[7],i[8],i[9],i[10],i[11],i[12],i[13],i[14],i[15],i[16],i[len(i)-2],i[len(i)-1]])
    for i in llistadecisiontree:
        i[-1]=i[-1].replace('\n','')
    finallist=[]
    for i in range(len(llistadecisiontree)):
        dict={}
        dict['Tipus']=float(llistadecisiontree[i][0])
        dict['Any']=float(llistadecisiontree[i][1])
        dict['Mobles']=float(llistadecisiontree[i][2])
        dict['Preu']=float(llistadecisiontree[i][3])
```

```
dict['Superficie']=float(llistadecisiontree[i][4])
dict['Habitacions']=float(llistadecisiontree[i][5])
dict['Banys']=float(llistadecisiontree[i][6])
dict['Aire']=float(llistadecisiontree[i][7])
dict['Terrassa']=float(llistadecisiontree[i][8])
dict['Balco']=float(llistadecisiontree[i][9])
dict['Garatge']=float(llistadecisiontree[i][10])
dict['Sauna']=float(llistadecisiontree[i][11])
dict['Gimnas']=float(llistadecisiontree[i][12])
dict['Ascensor']=float(llistadecisiontree[i][13])
dict['cee']=float(llistadecisiontree[i][14])
dict['class']=llistadecisiontree[i][16]
finallist.append(dict)
#####
#####
#####3
# A DTNUM.PY es troba la funcio GOODNESS IMPLEMENTADA
# GENEREM ARBRE DE DECISIO DE TOTA BARCELONA I CALCULEM EXACTITUD
dectree=DTnum.generateDT(finallist,max_depth=10,min_accuracy=0.8)

#####
#####
#####
#####PREPARACIO
DE LES NOSTRES DADES
PEL FLASK#####
#####
#####
#####
for i in district.values():
for j in range(len(i)):
i[j]=round(i[j],1)
diccionariiskmeans={}
#POSTANALISI
def analisi(postanalisi,distrito,dificultad):
counter0=[]
counter1=[]
counter2=[]
for i in postanalisi:
if i[0]==distrito and i[2]==dificultad:
if i[1]==1:
counter0.append(i)
elif i[1]==0:
counter1.append(i)
else:
counter2.append(i)
####j representa cadascun dels elements de qui volem fer la mitjana
estats=[]
if len(counter0)!=0:
for j in range(len(counter0[0][3])-2):
caracteristica=[]
for i in range(len(counter0)):
caracteristica.append(counter0[i][3][j])
if len(caracteristica)!=0:
estats.append((0,round(statistics.mean(caracteristica),1)))
estats.append(len(counter0))
if len(counter1)!=0:
for j in range(len(counter1[0][3])-2):
caracteristica=[]
for i in range(len(counter1)):
caracteristica.append(counter1[i][3][j])
if len(caracteristica)!=0:
estats.append((1,round(statistics.mean(caracteristica),1)))
estats.append(len(counter1))
if len(counter2)!=0:
for j in range(len(counter2[0][3])-2):
```

```
caracteristica=[]
for i in range(len(counter2)):
caracteristica.append(counter2[i][3][j])
if len(caracteristica)!=0:
stats.append((2,round(statistics.mean(caracteristica),1)))
stats.append(len(counter2))
return stats
for i in districtes:
for j in ['Fàcil','Mitjà','Difícil','No venut']:
diccionariskmeans[(i,j)]=analisi(postanalisi,i,j)

#####

#####

#####
#####INICIALITZACIO DEL FLASK UNA VEGADA QUE EL
PYTHON HA FET ELS SEUS
CALCULS#####
#####
#####
#####
app=Flask(__name__)
# ES DEFINEIX LA FUNCIO full_name PER PODER CRIDAR-LA DES DE LA
PLANTILLA seleccio.html QUE
DEFINEIX LA INTERACCIÓ DEL CLIENT AMB EL PRESENT PROJECTE
@app.context_processor
def some_processor():
def full_name(districte):
cucu=''
for i in llistakmeans:
if districte in i and 'No venut\n' in i :
cucu=cucu+str(i[0:-2])
return cucu
return {'full_name': full_name}
# ES DEFINEIX EL DECORADOR QUE ENS MOSTRA ELS ENLLAÇOS DELS DIFERENTS
DISTRICTES I QUE
SERVIRA PER ACCEDIR A LES ESTADÍSTIQUES I CENTROIDS DE CADA DISTRICTE.
FORMA PART
# DE LA PART DE L'EMPRESA.
@app.route('/')
def index():
return render_template("llistat.html",districtes=districtes)
# AQUEST DECORADOR ENLLAÇA LA PAGINA INICIAL QUE MOSTRA EL FLASK AMB
LES DADES DE CADA
DISTRICTE.
# FORMA PART DE L'EMPRESA
@app.route('/<districte>')
def barriada(districte):
return
render_template("barry.html",districte=districte,dictcentral=dictcentr
al,district=
district,camps=camps,diccionariskmeans=diccionariskmeans)
# AQUEST DECORADOR ENLLAÇA AMB LA PÀGINA DE FILTRES I RECOMANACIONS
PER AL CLIENT.
@app.route('/recomana')
def recomana():
return
render_template("seleccio.html",districtes=districtes,dictcentral=dict
central)
# CRIDA A LA CARREGA DE FLASK I LLENCAMENT AUTOMÀTIC DE LA WEB.
if __name__=="__main__":
location="http://127.0.0.1:5000"
webbrowser.open_new(location)
app.run(debug=True)
```

## 14.2. CODI FONT ARXIU DTNUM.PY

```
# -*- coding: utf-8 -*-
"""
Created on March 2016
@author: Andrés Cencerrado
"""

#Numeric DT implementation
####Activity 3: goodness function definition
def goodness(classes,attr_values):
    best_cutoff = (0,0,'','')
    #(cutoff value,proportion of correctly classified instances,majority
    class up to cutoff
    point,majority class from cutoff point)
    resultat=[]
    # CALCUL DELS PUNTS DE TALL. PRIMER ELIMINEM ELS VALORS REPETITS. ELS
    PUNTS DE TALL ES
    TROBEN A LA MEITAT EXACTE
    # ENTRE DOS VALORS DE UN ATRIBUT
    llistaatributsordered=list(set(sorted(attr_values)))
    classeatributs=list(set(classes))
    puntsdetall=[]
    for i in range(len(llistaatributsordered)-1):
        puntsdetall.append(((llistaatributsordered[i]+llistaatributsordered[i+1])/2.))
    # CALCULEM LA PROPORCIO DE CLASSE DE CADA PUNT DE TALL TAL COM SE
    EXPLICA A LA TEORIA.
    for punt in puntsdetall:
        comptadorsmenors=len(classeatributs)*[0]
        comptadorsmajors=len(classeatributs)*[0]
        for element in range(len(attr_values)):
            if attr_values[element]<punt:
                for k in range(len(classeatributs)):
                    if classes[element]==classeatributs[k]:
                        comptadorsmenors[k]=comptadorsmenors[k] + 1
            else:
                for k in range(len(classeatributs)):
                    if classes[element]==classeatributs[k]:
                        comptadorsmajors[k]=comptadorsmajors[k] + 1
        maxmenor=max(comptadorsmenors)
        maxmajor=max(comptadorsmajors)
        proporcio=(maxmenor+maxmajor)/float(len(attr_values))
        resultat.append((punt,proporcio,classeatributs[comptadorsmenors.index(
            maxmenor)],
            classeatributs[comptadorsmajors.index(maxmajor)]))
    resultatfinal=sorted(resultat, key=lambda x:x[1],reverse=True)
    # RETORNEM LA MILLOR TUPLA QUE RESULTA A SER EL PRIMER ELEMENT DESPRES
    DE ORDENAR ELS
    ELEMENTS PEL SEGON INDEX.
    best_cutoff=resultatfinal[0]
    return best_cutoff
#training_set = [{attr1: val1, attr2: val2, ... , 'class':class_id}]
#attribute named 'class' is mandatory
def generateDT(training_set, max_depth=5, min_accuracy=0.9):
    print("\nGenerating Decision Tree for a training set of " +
    str(len(training_set)) + "
    instances...")

    decision_tree = []
    training_subset1 = []
    training_subset2 = []
    classes = [instance['class'] for instance in training_set]
```



```
if classes.count(classes[0]) == len(classes): #if all instances in
training set have
same 'class' attribute
return ("class",classes[0])
attributes = {attribute:[instance[attribute] for instance in
training_set] for attribute
in training_set[0] if attribute != 'class'}
print("Obtaining goodness values... Remaining levels:
"+str(max_depth))
best_attribute = max([(attr,) + goodness(classes,attributes[attr]) for
attr in attributes
.keys()], key = lambda x: x[2]) #best attribute according to its
goodness
if (best_attribute[2] >= min_accuracy) | (max_depth == 1): #minimum
accuracy or
max_depth reached
return
[(best_attribute[0],best_attribute[1],("class",best_attribute[3]),("cl
ass",
best_attribute[4]))]
for i in range(len(training_set)):
if training_set[i][best_attribute[0]] <= best_attribute[1]:
training_subset1.append(training_set[i])
else:
training_subset2.append(training_set[i])
decision_tree.append((best_attribute[0],best_attribute[1],generateDT(t
raining_subset1,
max_depth=max_depth-
1,min_accuracy=min_accuracy),generateDT(training_subset2,max_depth=
max_depth-1,min_accuracy=min_accuracy)))
return decision_tree
def classify(instance, dectree):
if dectree[0] == 'class':
return dectree[1]
if instance[dectree[0][0]] < dectree[0][1]:
return classify(instance, dectree[0][2])
else:
return classify(instance, dectree[0][3])
def runTest(instance_list, dectree):
hits = 0
for i in instance_list:
real_class = i['class']
pred_class = classify(i,dectree)
if real_class == pred_class:

hits += 1
print('\n-----\nAccuracy: ' +
str(100*hits/len(instance_list)) +
'%\n-----')
```

### 14.3. CODI FONT ARXIU ESTADISTIQUES-TKINTER.PY

```
# -*- coding: utf-8 -*-
import io
from tkinter import *
import statistics
import numpy as np
import matplotlib.pyplot as plt
lb2=''
ax=''
dict={}
def curselet(evt):
```

```
global dict
global ax
plt.clf()
#value emmagatzema el nom del districte que clico a la listbox
value=str((Lb1.get(ANCHOR)))
any=[]
preu=[]
superficie=[]
habitacions=[]
banys=[]
garatge=[]
tipus=[]
mobles=[]
aire=[]
terrassa=[]
balco=[]
sauna=[]
gimnas=[]
ascensor=[]
cee=[]
subtipus=[]
estat=[]
piscina=[]
calefaccio=[]
camps=['any','preu
milers','superficie','habitacions','banys','garatge','% moblat','%
aire','terrassa','% balco','% sauna','% gimnas','% ascensor','% cee']
valors=[]
for i in llistadellistestotal:
if value in i:
any.append(int(i[7]))
preu.append(int(i[9].replace('.', '')))
superficie.append(int(i[11]))
habitacions.append(int(i[13]))
banys.append(int(i[15]))
garatge.append(int(i[19]))
terrassa.append(int(i[17]))
balco.append(int(i[18].replace('No','0')))
if i[8]=='Moblat':
i[8]='1'
else:
i[8]='0'
mobles.append(int(i[8]))
if i[16]=='No':
i[16]='0'
else:
i[16]='1'
aire.append(int(i[16]))
sauna.append(int(i[20]))
gimnas.append(int(i[21]))
if i[24]=='No':
i[24]='0'
else:
i[24]='1'
ascensor.append(int(i[24]))
if i[25]=='No':
i[25]='0'
else:
i[25]='1'
cee.append(int(i[25]))
if len(any)!=0 and len(preu)!=0 and len(superficie)!=0 and
len(habitacions)!=0:
valors.append(statistics.mean(any))
valors.append(statistics.mean(preu)/1000)
valors.append(statistics.mean(superficie))
valors.append(statistics.mean(habitacions))
valors.append(statistics.mean(banys))
```

```
valors.append(statistics.mean(garatge))
valors.append(100*statistics.mean(mobles))
valors.append(100*statistics.mean(aire))
valors.append(statistics.mean(terrassa))
valors.append(100*statistics.mean(balco))
valors.append(100*statistics.mean(sauna))
valors.append(100*statistics.mean(gimnas))
valors.append(100*statistics.mean(ascensor))
valors.append(100*statistics.mean(cee))
for j in range(len(valors)):
    valors[j]=round(valors[j],1)
pos = np.arange(len(camps))
width = 1.0 # gives histogram aspect to the bar diagram
rects2=plt.bar(pos, valors, width, color='r')
ax = plt.axes()
ax.set_xticks(pos + (width / 2))
ax.set_xticklabels(camps)
plt.title('MITJANES I PERCENTATGES: ' +str(value))
autolabel(rects2)
figManager = plt.get_current_fig_manager()
figManager.window.showMaximized()
plt.savefig(str(value)+'.png')
#plt.show(block=False)
plt.close()
def autolabel(rects):
    global ax
    # attach some text labels

    for rect in rects:
        height = rect.get_height()
        ax.text(rect.get_x() + rect.get_width()/2.,
            1.05*height,round(height,1),ha='center',
            va='bottom')
    #'%f'
    fitxer=io.open('BBDD-Ampliada-Nova.csv','r')
    llistastring=fitxer.readlines()
    llistadellistestotal=[]
    for i in llistastring:
        llistadellistestotal.append(i.split(';'))
    llistadellistestotal.remove(llistadellistestotal[0])
    top = Tk()
    Lbl = Listbox(top)
    Lbl.grid(row=1, column=4)
    districtes=[]
    for i in llistadellistestotal:
        if i[4] not in districtes and i[4] != '':
            districtes.append(i[4])
    for i in districtes:
        Lbl.insert(1, i)
    Lbl.bind('<<ListboxSelect>>',curselet)
    top.mainloop()
-
```

