



UNIVERSITAT_{DE}
BARCELONA

FACULTAT DE MATEMÀTIQUES I INFORMÀTICA

PROJECTE D'APRENENTATGE SERVEI

Intel·ligència Artificial



Noah Márquez Vara i Víctor Sort Rubio

Professors: Dra. Maite López-Sánchez i Dr. Ignasi Cos Aguilera

Contents

1	Introducció	2
2	Persones similars a tu	3
3	Anàlisi de dades	4
4	Curs de Machine Learning	11
5	Conclusions	12

1 Introducció

Inicialment, vàrem contactar amb les fundacions Pere Tarrés i Barcelona Open Data per veure quines propostes de treball ens oferien. Com vam veure que seria bastant complicat coordinar-nos bé amb ells per poder fer un projecte que ens beneficiés a ambdues parts, vam decidir finalment fer-lo de manera independent.

Així i tot, en veure la gran quantitat de repositoris de dades que oferia Barcelona Open Data, vam decidir crear una web usant els seus repositoris públics referents a dades estadístiques de la població de Barcelona. Barcelona Open Data és un moviment impulsat per les administracions públiques amb el principal objectiu d'aprofitar al màxim els recursos públics disponibles, exposant la informació generada o custodiada per organismes públics, permetent el seu accés i reutilització per al bé comú i per al benefici de persones i entitats interessades. Va néixer fa catorze anys i és gestionat des del Departament d'Estadística i Difusió de Dades de l'Oficina Municipal de Dades.

La web conté dues seccions principals. L'objectiu de la primera era crear una pantalla on la gent pugui interaccionar per saber quantes persones són similars a ells. En la segona es troben diverses petites seccions on es fan prediccions, es calculen esperances o es mostren gràfics sobre diferents dades. És una secció menys ordenada, però on hem pogut aplicar més eines de ML i aprendre més, a diferència de la primera. Es pot trobar fent click en el següent hiperlink: <https://dadesbarcelona.streamlit.app>.^{1 2}

A més a més, com explicarem més endavant, hem fet un curs de Machine Learning públic ofert per Google, que ens ha permès aprofundir amb els conceptes vistos a la classe de teoria de BigML.

¹Al ser un deploy gratuït, a l'hora de corregir el treball potser que s'hagi de reiniciar la web. Per qualsevol dubte se'ns pot contactar per email.

²Si es vol executar en local, serà necessari tenir Python instal·lat i, estant al directori del codi, fer un `pip install -r requirements.txt` i llavors `streamlit run home.py`.

2 Persones similars a tu

En aquesta primera secció de la pàgina web vem realitzar una mena de formulari per saber quantes persones que viuen a Barcelona compleixen les mateixes característiques que tu.

La pàgina web et va demanant diverses dades (sexe, edat, districte i barri de residència, lloc de naixement (continent, país i ciutat) i nivell educatiu) i pots decidir posar de totes elles les que vulguis. En cada cas, se't mostra quantes persones compleixen el mateix requisit en concret que tu, se t'obren potser noves possibles dades a inserir i, al final de la pàgina, s'actualitza el nombre total de persones que compleixen tots els requisits que hagis seleccionat.

Vàrem dedicar també una mica de temps a fer que la web tingués un disseny atractiu, acompanyant els textos de colors o diferents formats i emojis, i posant diverses explicacions.

A continuació es mostra com queda la web amb un exemple d'ús:

Introdueix les teves dades!
(O juga amb els valors per veure com varia el resultat...)

Sexe:
Home
Nombre de persones amb el mateix sexe que tu: 786183
Percentatge de persones amb el mateix sexe que tu: 47.558%

Edat:
21
-1 100
(Escull el valor -1 si no vols que es tingui en compte l'edat.)
(Escull el valor 100 si tens 100 o més anys.)
Nombre de persones amb la mateixa edat que tu: 16283
Percentatge de persones amb la mateixa edat que tu: 8.988%

Districte:
Sant Martí
Nombre de persones que viuen al mateix districte que tu: 241888
Percentatge de persones que viuen al mateix districte que tu: 14.638%

Barri:
el Clot
Nombre de persones que viuen al mateix barri que tu: 26486
Percentatge de persones que viuen al mateix barri que tu: 1.597%

Figure 1: Introduïnt dades (1)

Lloc de Naixement:
Resta de Catalunya
Nombre de persones que van neixer en la mateixa categoria que tu: 115535
Percentatge de persones que van neixer en la mateixa categoria que tu: 6.988%

Nivell Educatiu:
Estudis universitaris, CFGS grau superior
(Per edats menors a 16 anys no es tenen dades de nivell educatiu.)
Nombre de persones amb el mateix nivell educatiu que tu: 514573
Percentatge de persones amb el mateix nivell educatiu que tu: 31.122%

Amb les dades seleccionades...
Nombre de persones com tu: 4
Percentatge de persones com tu: 0.00024%

No està malament, has trobat uns pocs clons!

Figure 2: Introduïnt dades (2)

Per acabar aquest punt, comentar que gran part de la feina que hem hagut de fer, com ja ens havien advertit, va consistir en netejar dades. A més, tot i que a la web de Barcelona Open Data hi ha molts repositoris útils, entre ells a vegades hi ha diferències en la representació de les dades, fet que ens ha dificultat la tasca.

Per posar un exemple, en alguns repositoris es separava a la població per edat mentre que en altres s'agrupaven en grups de 5 o 10 anys. Per solucionar aquest problema, hem hagut de fer algunes assumpcions (en aquest cas per exemple, que en cada grup hi ha 1/5 part exacte de població amb cada una de les edats). Aquesta tasca es complicava bastant més quan s'havien de combinar fins a 5 repositoris entre ells. Vem donar-nos conta inclús que en tots els repositoris el total de població no era igual, hi ha una variancia de fins un 1% entre ells, que és un error que hem d'admetre, doncs no tenim encara les eines per solucionar-ho.

3 Anàlisi de dades

Predictor de Gènere Basat en Dades Demogràfiques de Barcelona

Aquesta part del projecte es centra en el desenvolupament d'un predictor de gènere basat en les dades demogràfiques proporcionades per OpenData Barcelona. Les dades inclouen informació sobre el districte, barri, edat i nivell educatiu de la població.

Càrrega i Preprocessament de Dades

Primerament, es carreguen diverses bases de dades que contenen informació demogràfica detallada de la població de Barcelona. Aquestes dades són essencials per entrenar el model de predicció. Es fan servir cinc conjunts de dades diferents, cadascun amb informació relativa a sexe, edat, nivell educatiu, i d'altres variables demogràfiques. Aquestes dades són processades i fusionades per crear un conjunt de dades consistent que es pot utilitzar per entrenar el model.

```
df_1 = analysis.load_data('datasets/2023_pad_mdba_sexe_edat-1.csv')
df_2 = analysis.load_data('datasets/2023_pad_mdb_lloc-naix_edat-q_sexe.csv')
df_3 = analysis.load_data('datasets/2023_pad_mdbas_lloc-naix-ccaa_sexe.csv')
df_4 = analysis.load_data('datasets/2023_pad_mdb_lloc-naix-continent_edat-q_sexe.csv')
df_5 = analysis.load_data('datasets/2023_pad_mdb_niv-educa-esta_edat-q_sexe.csv')
```

Entrenament del Model de Predicció de Gènere

Després de preprocessar les dades, procedim a entrenar el model de predicció de gènere. Es verifica si ja existeixen models previs; en cas contrari, s'entrena un nou model ³.

```
if os.path.exists('gender_models.pkl') ...
    gender_model, district_encoder, neighborhood_encoder =
        model.load_gender_models_and_encoders()
else
    X, y, df_merged, district_encoder, neighborhood_encoder = model.load_gender_model()
    gender_model = model.train_gender_model(X, y)
```

El model es basa en un *RandomForestClassifier* i s'entrena utilitzant les característiques seleccionades: districte, barri, grup d'edat i nivell educatiu.

El RandomForest és un algorisme d'aprenentatge supervisat que funciona construint múltiples arbres de decisió en el moment de l'entrenament i produint la classe (gènere en aquest cas) que és el mode de les classes (classificació) dels arbres individuals. S'adapta molt bé per a conjunts de dades amb un gran nombre de variables i pot controlar amb eficàcia les interaccions no lineals entre elles.

```
def train_gender_model(X, y):
    ...
    model = RandomForestClassifier(random_state=42)
    model.fit(X_train, y_train)
    ...
```

Interfície d'Usuari i Predicció

Finalment, el model es desplega mitjançant una interfície d'usuari, on els usuaris poden introduir dades com el districte, el barri, l'edat i el nivell educatiu. Basant-se en aquesta entrada, el model realitza una

³Al fer el deploy amb *streamlit*, la versió del deploy ja té els models carregats als seus servidors, per tant es poden fer servir directament. Si volguéssim obrir la web en local, hauriem d'esperar a que s'entrenés el model (≈ 4 segons) ja que el fitxer *pickle* ocupava massa per pujar-ho a *GitHub*.

predicció sobre el gènere de la persona. Aquesta aplicació pràctica demostra com les dades obertes i el Machine Learning poden ser utilitzats conjuntament per obtenir percepcions valuoses sobre la població.

```
district_options = district_encoder.classes_  
selected_district = st.selectbox('Districte', district_options)  
...
```

Finalment, el model realitza la predicció basant-se en les entrades de l'usuari.

```
if st.button('Prediu'):  
    prediction = gender_model.predict(...)  
    ...
```

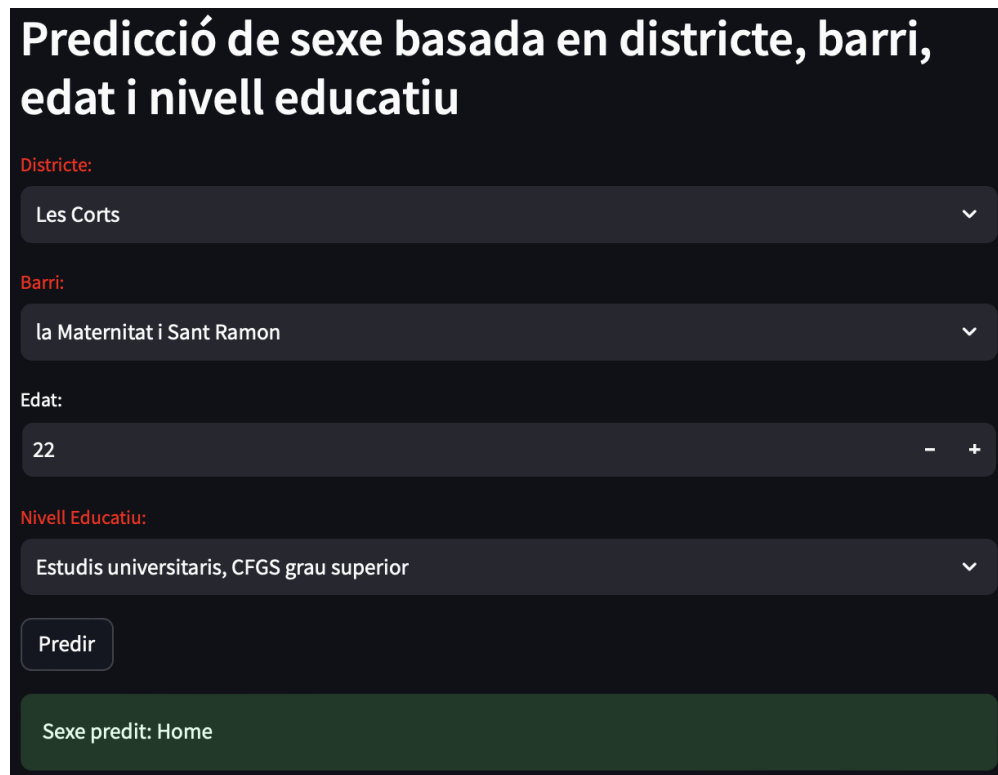


Figure 3: Interfície per la predicció de gènere

Predictor d'Esperança de Vida Basat en Dades Demogràfiques de Barcelona

La segona part del projecte implica el desenvolupament d'un predictor d'esperança de vida. Aquesta part analitza les dades relacionades amb el districte, barri, nivell educatiu i sexe per a predir l'interval d'edat en el qual es troba una persona.

Càrrega i Preparació de Dades

Al igual que en la primera part, es carreguen diversos conjunts de dades. Aquests conjunts de dades són fusionats basant-se en columnes comunes, com el nom del districte, el nom del barri, el grup d'edat quinquennal (EDAT_Q) i el sexe. Posteriorment, es realitza un procés d'enginyeria de característiques, on els noms dels districtes i barris són codificats utilitzant *LabelEncoder*.

```
def load_lifespan_model():
    # Load datasets
    df_main = pd.read_csv('datasets/2023_pad_mdb_niv-educa-esta_edat-q_sexe.csv')
    df_additional = pd.read_csv('datasets/2023_pad_mdb_lloc-naix_edat-q_sexe.csv')

    # Merging datasets on common columns
    df_merged = pd.merge(df_main, df_additional, on=['Nom_Districte', 'Nom_Barri', 'EDAT_Q',
    'SEXE'], how='inner')

    # Feature Engineering
    district_encoder = LabelEncoder()
    neighborhood_encoder = LabelEncoder()
    df_merged['Nom_Districte_Encoded'] =
    district_encoder.fit_transform(df_merged['Nom_Districte'])
    df_merged['Nom_Barri_Encoded'] =
    neighborhood_encoder.fit_transform(df_merged['Nom_Barri'])

    # Selecting relevant columns
    X = df_merged[['Nom_Districte_Encoded', 'Nom_Barri_Encoded', 'NIV_EDUCA_esta', 'SEXE']]

    return X, df_merged, district_encoder, neighborhood_encoder
```

Model de Predicció d'Esperança de Vida

Per a la predicció de l'esperança de vida, es construeixen múltiples models utilitzant *RandomForestClassifier*, un per a cada grup d'edat quinquennal. Aquests models s'entrenen per predir la probabilitat que una persona pertanyi a un determinat grup d'edat basant-se en les seves característiques codificades.

Les característiques utilitzades per a la predicció inclouen el districte codificat, el barri codificat, el nivell educatiu i el sexe. Es calcula l'exactitud de cada model per a validar la seva eficàcia.

```
def train_lifespan_models(X, df_merged):
    max_age_group = df_merged['EDAT_Q'].max()
    models = {}

    for age_group in range(1, max_age_group + 1):
        y = (df_merged['EDAT_Q'] == age_group).astype(int)
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
        random_state=42)
        model = RandomForestClassifier(random_state=42)
        model.fit(X_train, y_train)
        y_pred = model.predict(X_test)
        accuracy = accuracy_score(y_test, y_pred)
        print(f"Age Group {age_group}: Model Accuracy: {accuracy}")
        models[age_group] = model

    return models
```

Interfície d'Usuari per a la Predicció

Es proporciona una interfície on els usuaris poden seleccionar el seu sexe, districte, barri, edat i nivell educatiu. Aquests valors són codificats, i després utilitzats com a entrades per als models predictius.

```
...
# Age selection
age = st.number_input("Edat", min_value=0, max_value=100, value=25)
age_group = age // 5 # Mapping age to quinquennial group
...
```

Un cop l'usuari prem el botó de predicció, es calcula la probabilitat per a cada grup d'edat quinquennal, i es selecciona el grup amb la major probabilitat basant-se en les entrades de l'usuari. L'interval d'edat corresponent a aquest grup és llavors mostrat com a predicció de l'esperança de vida.⁴

```
def predict_lifespan(models, input_features):
    highest_probability = 0
    predicted_quinquennial_group = 1 # Default to the first group

    for age_group, model in models.items():
        probabilities = model.predict_proba([input_features])[0]

        # Handle the case where predict_proba returns only one column
        if len(probabilities) == 1:
            probability = 1 - probabilities[0]
        else:
            probability = probabilities[1]

        # Track the age group with the highest probability
        if probability > highest_probability:
            highest_probability = probability
            predicted_quinquennial_group = age_group
    # Ensure the predicted group is within valid range
    predicted_quinquennial_group = max(1, min(predicted_quinquennial_group,
max(models.keys())))

    # Convert the predicted quinquennial group to a normal age range
    predicted_age_group_min = (predicted_quinquennial_group - 1) * 5
    predicted_age_group_max = predicted_age_group_min + 4

    return predicted_age_group_min, predicted_age_group_max
```

Figure 4: Predicció de l'esperança de vida

⁴És important mencionar que per alguna combinació de dades escollides per l'usuari, no es tenen suficients dades. Llavors, el resultat de la predicció pot mostrar respostes no correctes, com per exemple 25 anys de predicció de vida.

Estudi de la Distribució de la Població

La tercera part del projecte implica la creació d'un gràfic de barres per mostrar la distribució de la població per districtes i barris de Barcelona. Aquesta secció visualitza les dades demogràfiques d'una manera clara i interactiva, proporcionant una visió general de la distribució de la població en diferents zones de la ciutat.

Estil i Configuració del Gràfic

S'utilitza la biblioteca *seaborn* per establir un estil visual per als gràfics, en aquest cas *whitegrid*, que proporciona un fons amb línies de graella per millorar la legibilitat. Es fa servir *altair*, una biblioteca de visualització de dades, per a construir el gràfic de barres.

```
# Custom styles for charts
sns.set(style="whitegrid")
```

Creació del Gràfic de Barres

La interfície permet a l'usuari seleccionar un districte a través d'un menú desplegable. Un cop seleccionat el districte, el codi filtra les dades per aquest districte específic i agrupa la població per barri.

El gràfic de barres és creat utilitzant *altair.Chart*, on l'eix X representa els barris (Nom.Barri), l'eix Y representa la població i cada barra té un color segons el barri.

S'inclouen eines de descripció emergent (*tooltip*) per mostrar informació addicional quan es passa el cursor sobre les barres.

El gràfic de barres resultant és mostrat en la interfície d'usuari amb l'ajuda de *st.altair_chart*. La visualització és interactiva, permetent als usuaris explorar les dades d'una manera més dinàmica.

```
bar_chart = altair.Chart(population_district).mark_bar().encode(
    x=altair.X('Nom_Barri:N', title=translations.translate('neighborhood_name')),
    y=altair.Y('Population:Q', title=translations.translate('population')),
    color='Nom_Barri:N',
    tooltip=[altair.Tooltip('Nom_Barri:N',
        title=translations.translate('neighborhood_name')),
        altair.Tooltip('Population:Q', title=translations.translate('population'))]
).interactive()

st.altair_chart(bar_chart, use_container_width=True)
```

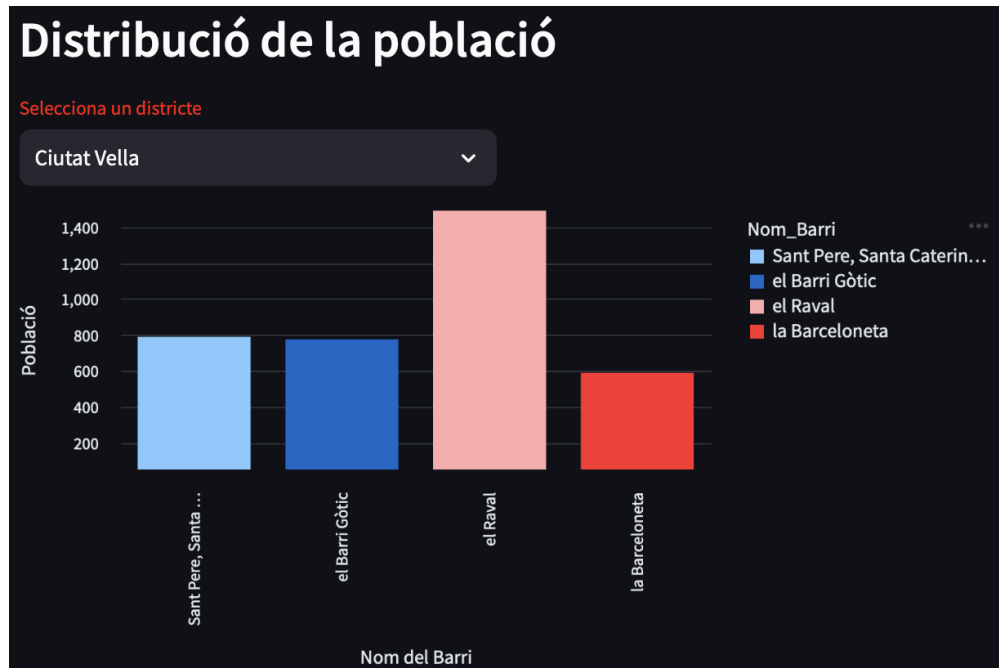


Figure 5: Distribució de la població segons districte

Resum Estadístic per Barri

La darrera secció del projecte consisteix en un estudi estadístic detallat de cada barri dins dels diferents districtes de Barcelona. Aquesta anàlisi proporciona un resum estadístic de la població, incloent mesures com l'edat mitjana, l'edat mediana, la desviació estàndard, l'edat mínima i màxima, així com els percentils 25 i 75.

Selecció de Districte i Barri

La interfície d'usuari permet seleccionar un districte i, basat en aquesta elecció, seleccionar un barri dins d'aquest districte. Aquesta selecció es realitza a través de menús desplegable, utilitzant `st.selectbox`.

```
district_stat = st.selectbox(translations.translate('select_district'),
df_1['Nom_Districte'].unique(), key='district_stat')
neighborhood_stat = st.selectbox(translations.translate('select_neighborhood'),
df_1[df_1['Nom_Districte'] == district_stat]['Nom_Barri'].unique(), key='neighborhood_stat')
```

Càlcul de l'Estadística Descriptiva

Un cop seleccionat un barri, s'extreu un subconjunt de dades corresponent a aquest barri específic. Es calcula un resum estadístic (`describe()`) d'aquestes dades, proporcionant informació sobre diferents mesures estadístiques relacionades amb l'edat de la població.

```
summary_stats = df_1[(df_1['Nom_Districte'] == district_stat) & (df_1['Nom_Barri'] ==
neighborhood_stat)]['EDAT_1'].describe()
```

Visualització del Resum Estadístic

El resum estadístic s'ensenyja en una targeta d'estil personalitzat mitjançant HTML i CSS incrustats. Cada mesura estadística (mitjana, mediana, desviació estàndard, mínim, màxim, percentils 25 i 75, i el nombre total) es presenta de manera clara, amb els seus valors corresponents.

```

<div class="stats-card">
  <h4>{translations.translate('summary_statistics')}</h4>
  ...
  <p><strong>{translations.translate('mean_age')}:</strong> {summary_stats['mean']:.2f}
  {translations.translate('years')}</p>
  ...
</div>

```

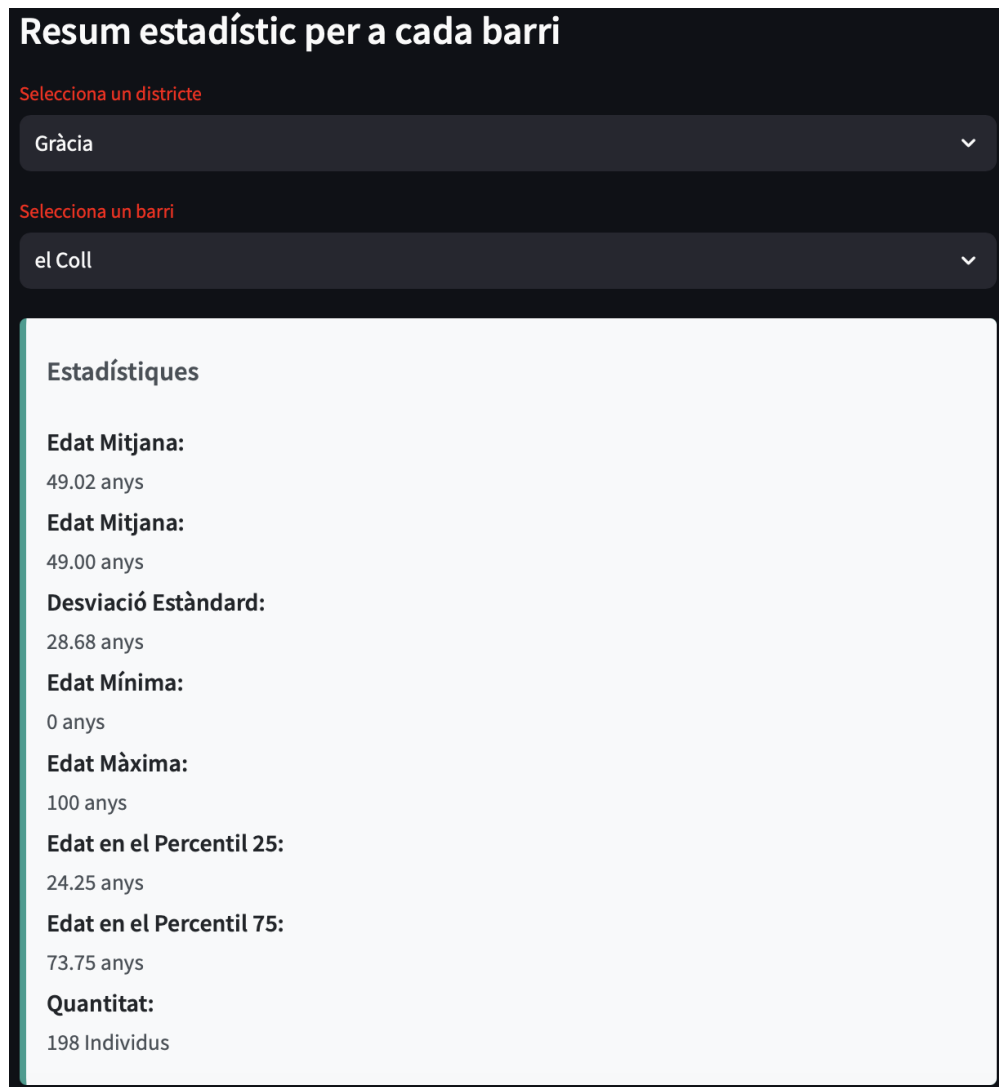


Figure 6: Resum estadístic per barri

4 Curs de Machine Learning

Els cursos que hem realitzat, proporcionats per Google for Developers, són els següents:

1. Introducció a l'aprenentatge automàtic (20 minuts)
2. Curs intensiu d'aprenentatge automàtic (15 hores)

i es poden trobar a la següent web: <https://developers.google.com/machine-learning?hl=es-419>

El primer és un curs molt curt on s'introdueixen els conceptes més bàsics referents a l'aprenentatge automàtic, molts dels quals ja coneixíem, doncs els hem tractat a l'assignatura.

El segon és un curs bastant més complet i aprofundeix en els següents temes:

- Regressió lineal.
- Entrenament i pèrdua (i descens de gradients).
- Tassa d'aprenentatge (i optimització d'aquesta) i sobreajustament.
- Conjunts d'entrenament, testeig i validació.
- Representació, neteja de dades i no linealitat.
- Regressió logística.
- Regularització (L0, L1 i L2).
- Càlcul de probabilitats.
- Pèrdua i regularització.
- Classificació. Cert vs Fals. Positiu vs Negatiu.
- Precisió i recuperació. Curva ROC i AUC.
- Xarxes Neuronals (amb múltiples classes).
- Incrustacions.

Un fet que ens ha agradat molt d'aquest curs és que justificaven matemàticament moltes de les expressions que donaven. Per posar un exemple, justificaven l'ús de les matrius Hessianes per calcular el descens de gradients.

A més a més, hi ha diversos tests als finals de cada lliçó per comprovar el coneixement assolit i diferents exercicis pràctics que estan molt bé. N'hi ha que són *Playground* en els quals s'ha de jugar amb molts dels hiperparàmetres que es presenten per controlar el procés d'aprenentatge i es veu de manera visual com l'elecció d'aquests afecta el resultat. N'hi ha d'altres que són Colabs de Google on se't guia per crear models a partir d'unes dades sobre preus de les cases a Califòrnia el 1990. A mesura que va avançant el curs, es va millorant el projecte, afegint hiperparàmetres, fent anàlisis dels resultats o inclús afegint-hi una xarxa neuronal.

5 Conclusions

La realització d'aquest treball i cursets ens ha permés aprofundir en uns coneixements que trobem molt interessants. De fet, els dos començarem a treballar en poc en empreses relacionades en el món de la intel·ligència artificial, el AA i el ML, així que ens ha sigut un treball molt profitós i que agraïm haver fet.

Repositori de GitHub

Treball-IA-repo

Llista de Paquets Utilitzats

Paquet	Descripció
streamlit	Eina per a crear aplicacions d'anàlisi de dades
matplotlib	Biblioteca per a la visualització de dades en Python
seaborn	Biblioteca basada en matplotlib per a gràfics estadístics
pandas	Biblioteca per a manipulació i anàlisi de dades
altair	Biblioteca de visualització de dades declarativa
scikit-learn	Biblioteca per a aprenentatge automàtic en Python
pickle	Mòdul per a serialització i deserialització d'objectes Python
RandomForestClassifier	Classe de scikit-learn per al mètode Random Forest

Table 1: Taula de paquets utilitzats en el projecte