

Mario Sánchez

**DISSENY D'UN SISTEMA INTEL·LIGENT PER A L'AVALUACIÓ DE LA
FATIGA A PARTIR DE DADES FISIOLÒGIQUES**

TREBALL DE FI DE GRAU

dirigit per Maria Ferré Bergadà

Grau d'Enginyeria Informàtica/Telemàtica



UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2025

Resum.

Aquest projecte desenvolupa un sistema intel·ligent capaç de predir el nivell de fatiga d'una persona a partir del senyal d'ECG, mitjançant una xarxa neuronal convolucional (CNN). La motivació neix de la necessitat d'avaluar la fatiga de forma objectiva i en temps real, especialment en àmbits mèdics i esportius. El sistema transforma el senyal temporal d'ECG en espectrogrames, que es classifiquen en 5 nivells de fatiga amb un model inspirat en l'arquitectura VGG.

Per preparar les dades d'entrada, s'apliquen tècniques de filtratge, normalització i remostreig, i s'utilitza la transformada de Fourier de temps curt (STFT) per convertir els senyals d'ECG en espectrogrames 2D. Aquesta representació temps-freqüència és especialment útil per capturar patrons relacionats amb l'estat fisiològic. Els resultats mostren que el model assoleix una precisió superior al 90% en dades conegudes i una capacitat de generalització prometedora amb dades pròpies i externes. A més, el sistema es complementa amb una aplicació mòbil multiplataforma per monitoritzar i predir la fatiga en temps real.

Resumen.

Este proyecto desarrolla un sistema inteligente capaz de predecir el nivel de fatiga de una persona a partir de la señal de ECG, mediante una red neuronal convolucional (CNN). La motivación surge de la necesidad de evaluar la fatiga de forma objetiva y en tiempo real, especialmente en contextos médicos y deportivos. El sistema transforma la señal temporal del ECG en espectrogramas, que se clasifican en 5 niveles de fatiga usando un modelo basado en la arquitectura VGG.

Para preparar los datos de entrada, se aplican técnicas de filtrado, normalización y remuestreo, y se utiliza la transformada de Fourier de tiempo corto (STFT) para convertir las señales de ECG en espectrogramas 2D. Esta representación tiempo-frecuencia es especialmente útil para captar patrones relacionados con el estado fisiológico. Los resultados muestran que el modelo alcanza una precisión superior al 90% con datos conocidos y presenta una capacidad de generalización prometedora con datos propios y externos. Además, el sistema se complementa con una aplicación móvil multiplataforma que permite la monitorización y predicción del nivel de fatiga en tiempo real.

Abstract.

This project develops an intelligent system capable of predicting a person's fatigue level based on ECG signals using a convolutional neural network (CNN). The motivation stems from the need to assess fatigue objectively and in real time, especially in medical and sports contexts. The system converts ECG signals into spectrograms, which are classified into 5 fatigue levels using a model based on the VGG architecture.

To prepare the input data, signal filtering, normalization, and resampling techniques are applied, and the Short-Time Fourier Transform (STFT) is used to convert ECG signals into 2D spectrograms. This time-frequency representation is particularly useful for capturing physiological patterns. Results show the model achieves over 90% accuracy on known data and demonstrates promising generalization with self-collected and external data. The system is also integrated into a multiplatform mobile app for real-time fatigue monitoring and prediction.

Índex

1	INTRODUCCIÓ	5
1.1	DESCRIPCIÓ GENERAL DEL PROJECTE	5
1.2	CONTEXTUALITZACIÓ	5
1.3	NECESSITATS DETECTADES	6
1.4	PREVISIÓ D'ÚS I USUARIS POTENCIALS	7
2	OBJECTIUS DEL PROJECTE	8
3	PLANIFICACIÓ	9
4	REQUISITS	10
4.1	REQUISITS FUNCIONALS.....	10
4.2	REQUISITS NO FUNCIONALS.....	16
4.2.1	<i>Requisits de la CNN</i>	16
4.2.2	<i>Requisits de l'aplicació</i>	17
5	ANÀLISI DELS REQUISITS FUNCIONALS	18
5.1	DIAGRAMA DE CASOS D'ÚS	18
5.2	DIAGRAMA DE CLASSES.....	19
6	DISSENY	20
6.1	PRECEDENTS.....	20
6.2	DADES INICIALS	20
6.3	PROCESSAMENT DEL SENYAL D'ENTRADA.....	21
6.3.1	<i>Filtratge del Senyal</i>	21
6.3.2	<i>Normalització</i>	22
6.3.3	<i>Segmentació del Senyal i Remostreig</i>	22
6.3.4	<i>Transformació a Espectrogrames mitjançant STFT</i>	22
6.4	ARQUITECTURA DEL MODEL.....	24
6.5	ADAPTACIÓ DEL MODEL VGG.....	25
6.5.1	<i>Fase Convolutiva (Extracció de Característiques)</i>	26
6.5.2	<i>Fase d'Aplanament</i>	26
6.5.3	<i>Fase de Classificació</i>	27
6.6	REDEFINICIÓ DELS NIVELLS DE FATIGA DE LES DADES INICIALS	27
6.7	DEFINICIÓ DELS CONJUNTS D'ENTRENAMENT, VALIDACIÓ I TEST	28
6.8	DISSENY DE L'APLICACIÓ MÒBIL	30
7	IMPLEMENTACIÓ	32
7.1	ENTORN DE DESENVOLUPAMENT	32
7.2	PROCESSAMENT DEL SENYAL	32
7.2.1	<i>Filtratge del Senyal</i>	32
7.3	IMPLEMENTACIÓ DE LA CNN.....	33
7.3.1	<i>Blocs convolucionals</i>	33
7.3.2	<i>Blocs fully connected</i>	33
7.4	ENTRENAMENT DEL MODEL	34
7.5	IMPLEMENTACIÓ DE L'APLICACIÓ MÒBIL	35
8	AVALUACIÓ	36
8.1	MÈTRIQES DE RENDIMENT GLOBAL	36
8.2	MÈTRIQES PER CLASSE	37
8.3	AVALUACIÓ DEL RENDIMENT SOBRE MOSTRES DIFERENTS A LES DADES INICIALS	39
9	CONCLUSIONS	41
10	CONSIDERACIONS ÈTIQUES I DE RESPONSABILITAT SOCIAL	43
10.1	IGUALTAT	43

10.2	MEDI AMBIENT	43
10.3	RESPONSABILITAT SOCIAL.....	43
10.4	ÈTICA.....	43
11	REFERÈNCIES.....	44

Índex de taules

TAULA 1. RESUM DE LES MÈTRIQVES PER CLASSE.....	38
TAULA 2. RESULTATS EXPERIMENTALS DE LES PREDICCIONS EN DIFERENTS CONDICIONS.....	40

Índex de figures

FIGURA 1. FORMA D'UN ECG QUE MOSTRA CADASCUN DELS PICS CORRESPONENTS A UN BATEC DEL COR.....	6
FIGURA 2. DIAGRAMA DE GANTT DEL DESENVOLUPAMENT DEL PROJECTE.	9
FIGURA 3. DIAGRAMA DE CASOS D'ÚS.....	18
FIGURA 4. DIAGRAMA DE CLASSES.....	19
FIGURA 5. ESPECTROGRAMES OBTINGUTS DESPRÉS DEL PROCESSAMENT DELS SENYALS D'ECG.....	24
FIGURA 6. ARQUITECTURA DEL MODEL VGG16.....	25
FIGURA 7. EVOLUCIÓ DE LA FORMA DELS ESPECTROGRAMES EN LES ETAPES DE CONVOLUCIÓ.....	26
FIGURA 8. EVOLUCIÓ DE LES MÈTRIQUES D'ENTRENAMENT I VALIDACIÓ SEGONS LA COMBINACIÓ D'HIPERPARÀMETRES.	30
FIGURA 9. Matriu de confusió de les prediccions del model sobre el conjunt de test.	36
FIGURA 10. AUC per cadascun dels nivells de fatiga.	38

1 Introducció

1.1 Descripció General del Projecte

Aquest projecte parteix d'una idea senzilla però amb molt potencial: entendre com el nostre cos ens parla quan està cansat, i fer-ho mitjançant la IA¹. L'objectiu central és desenvolupar un model capaç de predir automàticament el nivell de fatiga d'una persona, en una escala de l'1 al 5, a partir del senyal d'un ECG² prèviament processat. Per aconseguir-ho, s'ha fet servir una CNN³, un tipus de model de DL⁴ especialment eficaç en el reconeixement d'imatges i la seva classificació [1].

Un dels grans reptes —i alhora una de les forteses— del projecte és que el model ha estat entrenat amb dades recollides amb un sensor diferent i a una freqüència de mostreig diferent de la que s'utilitza en el desenvolupament pràctic. Aquesta diferència fa que el model hagi de ser especialment robust i adaptable: es vol que sigui generalitzable, capaç de funcionar bé més enllà del conjunt de dades original. Per tal de validar aquesta capacitat, s'han registrat i analitzat mostres pròpies en diversos moments del dia i en diferents estats de fatiga, posant a prova la sensibilitat del model per captar canvis subtils i reals en el nivell de cansament. També s'han usat mostres de ciclistes que col·laboren amb ISGlobal, grup amb el qual he tingut l'oportunitat de col·laborar i que m'ha brindat el suport necessari per desenvolupar aquest treball de final de grau.

Tot i que l'enfocament del projecte és essencialment tècnic, també s'ha pensat en la seva aplicació real. El model es vol aplicar principalment en dos àmbits: per una banda, en el seguiment de dones que estan rebent tractament contra el càncer, ajudant a detectar i entendre millor com evoluciona la seva fatiga. D'altra banda, també es vol utilitzar sobre ciclistes d'alt rendiment, que necessiten controlar l'esforç i optimitzar la recuperació. A més llarg termini, s'estudia la possibilitat d'integrar aquest model en una aplicació mòbil multiplataforma com a eina de suport i visualització de resultats. Ara bé, aquest desenvolupament es considera complementari, ja que el focus principal del projecte se centra en la investigació i validació del model d'IA.

Paraules clau: fatiga, xarxa neuronal convolucional (CNN), electrocardiograma (ECG), processament de senyals, intel·ligència artificial (IA).

1.2 Contextualització

La fatiga és un estat fisiològic caracteritzat per una disminució temporal de la capacitat del cos per rendir al seu nivell habitual, resultant d'una activitat física o mental prolongada i exigent. Aquest concepte es descriu també com una percepció o emoció, un mecanisme important per minimitzar el risc de lesions físiques, així com un símptoma, un risc, una causa o una conseqüència [2].

¹ Intel·ligència Artificial

² Electrocardiograma

³ *Convolutional Neural Network*. La gran majoria de termes estan en notació anglesa. Per cada terme utilitzat, es fa servir la seva notació més habitual en aquest entorn.

⁴ *Deep Learning*

La multiplicitat de definicions i conceptualitzacions de la fatiga s'explica per la seva àmplia presència i dispersió en diferents camps d'investigació i disciplines professionals. Concretament, en l'àmbit mèdic i esportiu, la fatiga es manifesta sovint com una reducció de la concentració, l'energia o el rendiment general, amb implicacions rellevants tant en la salut com en la seguretat de les persones [2]. Tanmateix, tot i ser un fenomen àmpliament estudiat, encara existeixen dubtes sobre la seva definició precisa i els mecanismes subjacents, fet que ha impulsat contínuament noves propostes de redefinició [3].

Tot i la seva rellevància, la fatiga continua sent una variable complexa de mesurar, especialment de forma objectiva i en temps real. Tradicionalment, s'ha quantificat mitjançant indicadors subjectius, com escales de percepció de l'esforç, o proves de rendiment, com la pèrdua de velocitat o de força [4]. No obstant això, l'aparició de tecnologies avançades com dispositius *wearable* i les tècniques de tractament de senyals fisiològics obren oportunitats sense precedents per mesurar la fatiga d'una manera no invasiva, contínua i objectiva [5].

En aquest context, els senyals cardíacs, com l'ECG, emergeixen com una font d'informació rica per avaluar l'estat fisiològic de l'individu [5]. Paràmetres derivats d'aquests senyals, com la variabilitat de la freqüència cardíaca (VFC), han esdevingut eines prometedores per avaluar la salut cardiovascular. Concretament, la VFC, que es defineix com la variació de temps entre 2 batecs de cor consecutius (també es pot definir com la distància corresponent a l'interval R-R, com es pot veure en la *Figura 1*), ha demostrat ser útil en l'avaluació de l'estrès i la fatiga [6], [7].

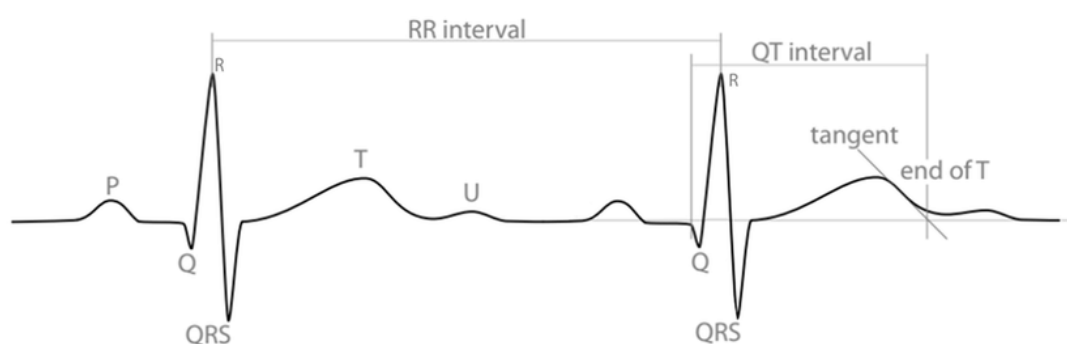


Figura 1. Forma d'un ECG que mostra cadascun dels pics corresponents a un batec del cor.

Tanmateix, la IA aplicada a l'anàlisi de l'ECG ha demostrat un potencial significatiu a l'hora de diagnosticar i monitoritzar malalties cardiovasculars. **Algoritmes d'IA** permeten analitzar la forma completa d'un ECG, oferint **informació molt més precisa i comprensiva** que els mètodes tradicionals com l'anàlisi dels intervals R-R. Particularment, l'ús de DL, concretament CNNs, han demostrat tenir una capacitat de diagnosi de condicions cardíacques comparable a la d'especialistes [8].

Així, el desenvolupament d'un model d'IA capaç de predir el nivell de fatiga a partir de la forma de l'ECG representa una oportunitat per proporcionar una eina objectiva per avaluar la fatiga, amb aplicacions potencials en l'àmbit mèdic i esportiu.

1.3 Necessitats Detectades

La necessitat d'eines que permetin monitoritzar la fatiga de manera objectiva i contínua és cada vegada més evident. En un món on la salut personalitzada i el rendiment òptim són cada cop més rellevants, detectar la fatiga de forma precoç es converteix en una prioritat en molts àmbits. Des de la cura de pacients que estan en tractament contra el càncer fins a

L'optimització del rendiment d'esportistes d'elit, les aplicacions d'aquestes eines són diverses i essencials.

En el món de la **medicina**, la detecció primerenca de la fatiga pot marcar la diferència entre una resposta eficaç al tractament i complicacions greus. Pacients que reben teràpies intensives necessiten eines que no només els permetin seguir els símptomes més evidents, sinó que també detectin canvis subtils en el seu estat físic que puguin indicar riscos o la necessitat de reajustar el tractament.

En l'àmbit **esportiu**, la capacitat de controlar la fatiga amb precisió pot ser decisiva per aconseguir el millor rendiment d'un atleta. Saber quan ajustar la intensitat i el tipus d'entrenament segons el nivell de fatiga pot ser clau per millorar els seus resultats i evitar el sobreentrenament.

L'ús d'IA i altres tecnologies emergents permeten no només monitoritzar de manera remota, sinó també analitzar en profunditat les senyals fisiològiques. Aquesta capacitat d'interpretar la complexitat dels senyals cardíacs pot transformar la manera en què gestionem la fatiga, millorant la qualitat de vida dels pacients i optimitzant el rendiment dels esportistes.

1.4 Previsió d'Ús i Usuaris Potencials

Aquest projecte té un enfocament dual, amb beneficis clars per a dos grups clau:

- **Dones amb càncer:** Permetrà monitoritzar com diferents tractaments afecten la seva fatiga, millorant la qualitat de vida i l'activitat diària de manera personalitzada.
- **Ciclistes professionals i d'alt rendiment:** En col·laboració amb ISGlobal, ajudarà a optimitzar la seva càrrega d'entrenament, prevenir el sobreentrenament i reduir el risc de lesions.

Aquesta combinació d'aplicacions en àmbits mèdics i esportius demostra la versatilitat del sistema, amb possibilitats de futura expansió cap a altres àmbits prometedors.

2 Objectius del Projecte

Objectiu General

Desenvolupar i validar un model d'IA innovador, basat en una CNN, capaç de predir objectivament el nivell de fatiga utilitzant la forma del senyal d'ECG, tot assegurant-ne la robustesa i la capacitat de generalització a diferents dispositius i condicions de captura de dades, mitjançant tècniques avançades d'anàlisi de senyals i ML⁵.

Objectius Específics

- Revisar i analitzar estudis existents sobre models predictius de fatiga basats en característiques extrems de l'ECG o de la mateixa senyal ECG.
- Desenvolupar un sistema robust de processament de senyals ECG que asseguri una entrada consistent i fiable per al model, independentment del dispositiu utilitzat per obtenir els senyals.
- Construir i entrenar un model predictiu avançat que sigui capaç de reconèixer patrons rellevants del senyal ECG processat i classificar amb precisió els diferents nivells de fatiga.
- Validar experimentalment la capacitat del model per generalitzar, mitjançant dades pròpies recollides amb un dispositiu diferent del del conjunt d'entrenament, i utilitzant mètriques com l'exactitud, la precisió, la sensibilitat o l'àrea sota la corba ROC⁶.
- Avaluar l'estabilitat i la sensibilitat del model davant canvis subtils en l'estat de fatiga, mitjançant una anàlisi crítica dels resultats obtinguts amb mostres registrades en diferents moments i condicions individuals.
- Identificar els punts forts i febles del model desenvolupat, i proposar línies de millora per a futures iteracions o aplicacions clíniques/esportives més específiques.
- *Integrar, de manera complementària, el model desenvolupat en una aplicació mòbil multiplataforma intuïtiva, que permeti als usuaris finals monitoritzar fàcilment i de manera autònoma el seu estat de fatiga.*

Aquest projecte té com a finalitat oferir una solució innovadora que combini de manera efectiva tecnologia, salut i esport, generant un impacte positiu directe tant en la qualitat de vida dels pacients com en el rendiment i la seguretat dels esportistes.

⁵ *Machine Learning*

⁶ *Receiver Operating Characteristic*

3 Planificació

La següent *Figura* mostra el digrama de Gantt representat com s'ha planificat el desenvolupament d'aquest projecte.

GANTT CHART

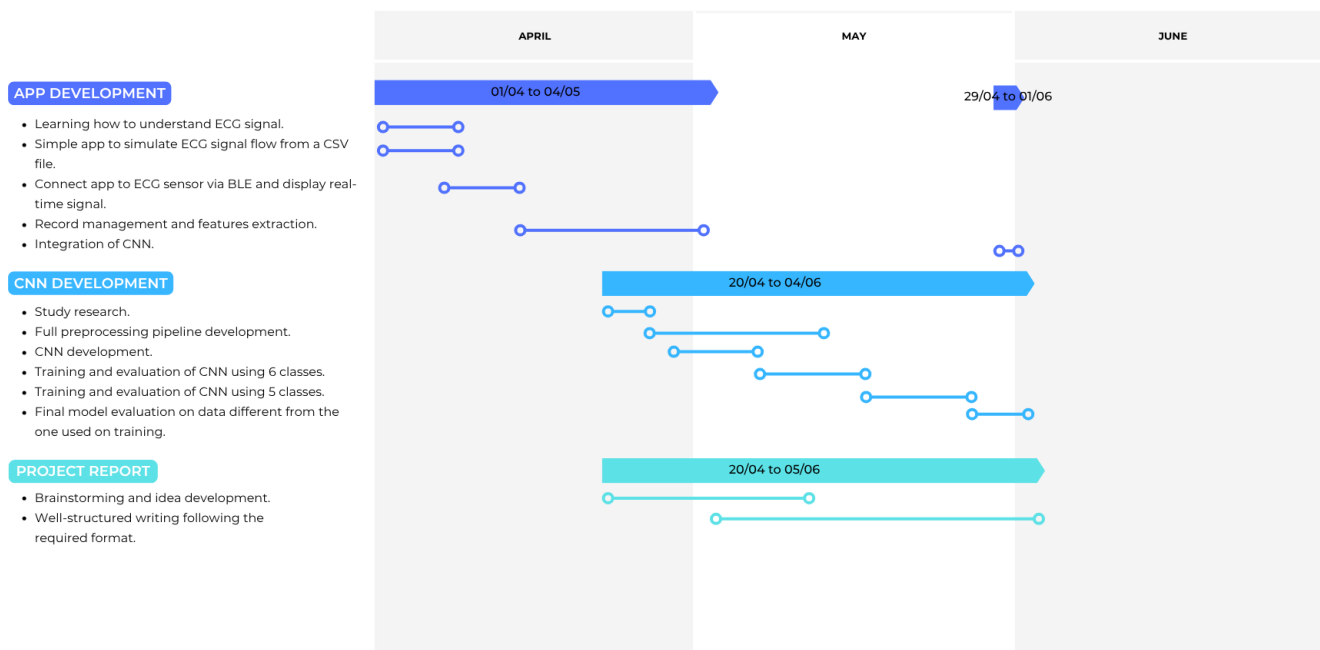


Figura 2. Diagrama de Gantt del desenvolupament del projecte.

4 Requisits

4.1 Requisits Funcionals

Com ja s'ha comentat, el focus d'aquest projecte es centra en el desenvolupament del model predictiu. Tanmateix, amb la possibilitat d'integrar-lo a una aplicació mòbil, a continuació es detallen breument les diferents funcionalitats d'aquesta aplicació mitjançant la descripció dels seus casos d'ús, corresponents a una **versió primerenca de l'aplicació**:

00. *Send Single Data*⁷

Resum funcionalitat: envia un sol conjunt de dades⁸ al sistema.

Paràmetres entrada: Nom del fitxer CSV amb les dades del senyal d'ECG i les característiques extretes a partir d'aquest.

Paràmetres sortida: codi de resposta.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: el sistema ha rebut el conjunt de dades.

Procés normal principal:

1. L'usuari selecciona quin enregistrament vol enviar.
2. 02. *Send Data*.

Alternatives de procés i excepcions: cap.

01. *Send Multiple Data*

Resum funcionalitat: envia diferents conjunts de dades de cop al sistema.

Paràmetres entrada: Llista amb els noms dels diferents fitxers CSV, cadascun amb les dades del senyal d'ECG i les característiques extretes a partir d'aquest.

Paràmetres sortida: codi de resposta.

Actors: Usuari.

Precondició: almenys existeix un enregistrament (també es pot seleccionar aquesta opció i triar un únic enregistrament).

Postcondició: el sistema ha rebut els diferents conjunts de dades.

Procés normal principal:

1. L'usuari selecciona quins enregistraments vol enviar.
2. 02. *Send Data*.

Alternatives de procés i excepcions: cap.

02. *Send Data*

Resum funcionalitat: gestiona de manera general l'enviament de dades.

Paràmetres entrada: llista amb els noms dels fitxers CSV a enviar.

Paràmetres sortida: codi de resposta.

Actors: Usuari i Servidor.

Precondició: almenys existeix un enregistrament.

Postcondició: el sistema ha rebut les dades.

Procés normal principal:

⁷ Tot el codi del projecte s'ha desenvolupat en anglès, és per això que algunes notacions i Figures consten així.

⁸ Un conjunt de dades fa referència a una gravació/enregistrament, és a dir, a les dades corresponents al senyal d'ECG, juntament amb les possibles característiques extretes del propi senyal.

1. L'aplicació selecciona el/s fitxer/s CSV guardat/s en el seu emmagatzematge intern.
2. L'aplicació envia el/s fitxer/s CSV via REST⁹ a un **servidor local**.
3. L'aplicació rep el missatge de confirmació del servidor i el mostra a l'usuari.

Alternatives de procés i excepcions:

- 2a. Error en l'enviament: es mostra un missatge d'error.

03. Download Single Data

Resum funcionalitat: descarrega un sol conjunt de dades.

Paràmetres entrada: Nom del fitxer CSV amb les dades del senyal d'ECG i les característiques extreteres a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: el conjunt de dades ha sigut descarregat.

Procés normal principal:

1. L'usuari selecciona quin enregistrament vol descarregar.
2. 05. Download Data.

Alternatives de procés i excepcions: cap.

04. Download Multiple Data

Resum funcionalitat: descarrega diferents conjunts de dades de cop.

Paràmetres entrada: Llista amb els noms dels diferents fitxers CSV, cadascun amb les dades del senyal d'ECG i les característiques extreteres a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament (també es pot seleccionar aquesta opció i triar un únic enregistrament).

Postcondició: els conjunts de dades han sigut descarregats.

Procés normal principal:

1. L'usuari selecciona quins enregistraments vol descarregar.
2. 05. Download Data.

Alternatives de procés i excepcions: cap.

05. Download Data

Resum funcionalitat: gestiona de manera general la descàrrega de dades.

Paràmetres entrada: llista amb els noms dels fitxers CSV a descarregar.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: les dades han estat descarregades.

Procés normal principal:

1. L'aplicació selecciona el/s fitxer/s CSV guardat/s en el seu emmagatzematge intern.
2. L'aplicació fa una crida al sistema operatiu que està executant l'aplicació, i aquest mateix s'encarrega de gestionar la descàrrega.

⁹ REpresentational State Transfer

Alternatives de procés i excepcions: cap.

06. Rename Data

Resum funcionalitat: permet canviar el nom d'un enregistrament o gravació.

Paràmetres entrada: Nom del fitxer CSV amb les dades del senyal d'ECG i les característiques extretes a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: l'enregistrament ha estat renombrat.

Procés normal principal:

1. L'usuari selecciona quin enregistrament vol renombrar.
2. L'aplicació mostra un *pop-up* que demana un nou nom per l'enregistrament.
3. L'usuari escriu un nom i prem *Ok*.
4. L'aplicació actualitza el nom del fitxer CSV i una taula de metadades guardada a la seva pròpia base de dades.

Alternatives de procés i excepcions:

3a. L'usuari prem *Cancel*: l'aplicació no fa cap actualització.

3b. L'usuari escriu un nom ja existent i prem *Ok*: l'aplicació no fa cap actualització i mostra missatge d'error.

3c. L'usuari no escriu res i prem *Ok*: l'aplicació actualitza el nom del fitxer CSV i la taula de metadades amb un nom predeterminat.

07. Delete Single Data

Resum funcionalitat: elimina un sol conjunt de dades.

Paràmetres entrada: Nom del fitxer CSV amb les dades del senyal d'ECG i les característiques extretes a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: el conjunt de dades ha estat eliminat.

Procés normal principal:

1. L'usuari selecciona quin enregistrament vol eliminar.
2. 09. Delete Data.

Alternatives de procés i excepcions: cap.

08. Delete Multiple Data

Resum funcionalitat: elimina diferents conjunts de dades de cop.

Paràmetres entrada: Llista amb els noms dels diferents fitxers CSV, cadascun amb les dades del senyal d'ECG i les característiques extretes a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament (també es pot seleccionar aquesta opció i triar un únic enregistrament).

Postcondició: els conjunts de dades han sigut eliminats.

Procés normal principal:

1. L'usuari selecciona quins enregistraments vol eliminar.
2. 09. Delete Data.

Alternatives de procés i excepcions: cap.

09. Delete Data

Resum funcionalitat: gestiona de manera general l'eliminació de dades.

Paràmetres entrada: llista amb els noms dels fitxers CSV a eliminar.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: les dades han estat eliminades.

Procés normal principal:

1. L'aplicació selecciona el/s fitxer/s CSV guardat/s en el seu emmagatzematge intern i l'/els elimina.
2. L'aplicació actualitza la taula de metadades eliminant les files corresponents al/s fitxer/s.

Alternatives de procés i excepcions: cap.

10. Show Detailed Data of Record

Resum funcionalitat: mostra la informació detallada d'un enregistrament. Això inclou el **nivell de fatiga**.

Paràmetres entrada: Nom del fitxer CSV amb les dades del senyal d'ECG i les característiques extrems a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: almenys existeix un enregistrament.

Postcondició: cap.

Procés normal principal:

1. L'aplicació selecciona el fitxer CSV guardat en el seu emmagatzematge intern.
2. L'aplicació mostra el **nivell de fatiga predit**.
3. L'aplicació mostra el senyal d'ECG en un gràfic de tipus *scrollable*.
4. L'aplicació mostra l'evolució de $DFA-\alpha I^{10}$ en un gràfic de tipus *scrollable*.
5. L'aplicació mostra els intervals que mostra la *Figura 1* en diferents gràfics de barres.

Alternatives de procés i excepcions:

- 2a. El nivell de fatiga encara no està predit: l'aplicació mostra un botó per dur a terme la predicció.

11. Display real-time ECG

Resum funcionalitat: mostra el senyal d'ECG en temps real quan el sensor està connectat.

Paràmetres entrada: cap.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: el sensor ha d'estar connectat.

Postcondició: cap.

Procés normal principal:

1. L'aplicació es subscriu al servei d'ECG del sensor.
2. L'aplicació rep les dades del senyal d'ECG i les dibuixa a la pantalla aconseguint el moviment típic d'un ECG.

¹⁰ *Detrended Fluctuation Analysis - αI*

3. Torna al pas 2.

Alternatives de procés i excepcions:

2a. El sensor es desconnecta: l'aplicació atura el dibuix del senyal i torna a mostrar la pantalla principal on s'indica que no hi ha cap sensor connectat.

12. **Start Recording (Calculate ECG Features)**

Resum funcionalitat: comença l'enregistrament i calcula dinàmicament característiques del senyal d'ECG.

Paràmetres entrada: cap.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: el senyal d'ECG s'ha d'estar mostrant.

Postcondició: cap.

Procés normal principal:

1. L'aplicació mostra un *pop-up* que demana un nom per l'enregistrament.
2. L'usuari escriu un nom i prem *Ok*.
3. L'aplicació crea una llista buida per acumular les dades procedents del sensor.
4. L'aplicació troba els pics R del senyal d'ECG dins les dades de la llista.
5. A partir dels pics R, l'aplicació calcula els intervals que mostra la *Figura 1*, i un paràmetre anomenat *DFA- $\alpha 1$* .
6. L'aplicació guarda un històric del senyal d'ECG i de les seves característiques.
7. L'aplicació actualitza els valors que es mostren per pantalla.
8. Als 5 segons, torna al pas 3.

Alternatives de procés i excepcions:

2a. L'usuari prem *Cancel*: l'aplicació cancel·la el cas d'ús.

2b. L'usuari escriu un nom ja existent i prem *Ok*: l'aplicació afegeix un *timestamp* per evitar guardar un nom duplicat.

2c. L'usuari no escriu res i prem *Ok*: l'aplicació crea un nom per defecte.

13. **Stop Recording (Calculate ECG Features)**

Resum funcionalitat: atura l'enregistrament i emmagatzema les característiques extretes del senyal d'ECG.

Paràmetres entrada: històric del senyal d'ECG i de les seves característiques.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: el sensor ha d'estar connectat.

Postcondició: s'ha creat un fitxer CSV i una nova fila a la taula de metadades.

Procés normal principal:

1. L'aplicació atura l'enregistrament (12. *Start Recording (Calculate ECG Features)*) del senyal d'ECG i el càlcul de característiques.
2. L'aplicació crea un fitxer CSV amb el nom generat en el cas d'ús anterior, i afegeix una fila a la taula de metadades de la base de dades.

Alternatives de procés i excepcions: cap.

14. **Predict Fatigue Level**

Resum funcionalitat: **prediu el nivell de fatiga.**

Paràmetres entrada: Nom del fitxer CSV amb les dades del senyal d'ECG i les característiques extretes a partir d'aquest.

Paràmetres sortida: cap.

Actors: Usuari i Servidor.

Precondició: almenys existeix un enregistrament.

Postcondició: **s'ha predit el nivell de fatiga.**

Procés normal principal:

1. L'aplicació mostra un botó per dur a terme la predicció del nivell de fatiga.
2. L'usuari prem el botó i es mostra un *pop-up* que indica que s'està realitzant la predicció.
3. L'aplicació envia el fitxer CSV via REST a un **servidor local**.
4. El servidor retorna la **predicció del nivell de fatiga**.
5. L'aplicació afegeix la predicció a la taula de metadades i actualitza la pantalla amb la informació detallada per aquest enregistrament.

Alternatives de procés i excepcions:

- 3a. Error en l'enviament: es mostra un missatge d'error.

15. *Connect To Sensor*

Resum funcionalitat: connecta un sensor a l'aplicació via *Bluetooth*.

Paràmetres entrada: nom del sensor i el seu número de sèrie.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: l'opció *Bluetooth* del mòbil està connectada.

Postcondició: l'aplicació està connectada al sensor.

Procés normal principal:

6. L'aplicació mostra una llista dels dispositius *Movesense* disponibles, que permeten una connexió a través de BLE¹¹.
7. L'usuari tria un dispositiu.
8. L'aplicació es connecta al sensor triat.
9. L'aplicació actualitza una altra taula de metadades, que conté informació dels sensors, de la base de dades.

Alternatives de procés i excepcions:

- 2a. L'usuari prem *Cancel*: l'aplicació no continua amb el procés de connexió.

- 3a. La connexió no es pot dur a terme: l'aplicació mostra un missatge d'error i acaba amb el procés de connexió.

16. *Disconnect From Sensor*

Resum funcionalitat: desconnecta el sensor que està en aquest moment connectat.

Paràmetres entrada: nom del sensor i el seu número de sèrie.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: el sensor està connectat.

Postcondició: l'aplicació deixa d'estar connectada al sensor.

Procés normal principal:

1. L'aplicació es desconnecta del sensor.

Alternatives de procés i excepcions:

- 1a. L'aplicació està executant *I2. Start Recording (Calculate ECG Features)*: l'aplicació mostra un missatge d'error informant de la situació.

¹¹ *Bluetooth Low Energy*

17. Forget Sensor

Resum funcionalitat: oblida (elimina la vinculació) amb un sensor.

Paràmetres entrada: nom del sensor i el seu número de sèrie.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: el sensor ha estat connectat alguna vegada.

Postcondició: el sensor ha estat oblidat.

Procés normal principal:

1. L'aplicació elimina la informació del sensor de la taula de metadades corresponents als sensors.

Alternatives de procés i excepcions:

- 1a. El sensor està connectat: l'aplicació executa *14. Disconnect From Sensor* i torna al pas 1.

18. Change Theme Mode

Resum funcionalitat: canvia el tema de la interfície (clar/fosc).

Paràmetres entrada: booleà que indica quin tema és l'actual.

Paràmetres sortida: cap.

Actors: Usuari.

Precondició: cap.

Postcondició: l'aplicació ha canviat de tema.

Procés normal principal:

1. L'aplicació actualitza la variable booleana (la inverteix)
2. L'aplicació refresca el tema.

Alternatives de procés i excepcions: cap.

4.2 Requisits No Funcionals

Tot i que habitualment els requisits no funcionals es defineixen exclusivament per a aplicacions, també resulta interessant aplicar aquest tipus de requisits al disseny de la CNN, ja que la qualitat del model predictiu dependrà en gran mesura de factors no funcionals.

4.2.1 Requisits de la CNN

- **Precisió desitjada:** la CNN ha de garantir un nivell elevat de precisió per assegurar que les prediccions sobre el nivell de fatiga siguin fiables.
- **Facilitat d'ús:** la xarxa neuronal ha de ser fàcil d'integrar en altres sistemes o aplicacions. Tot i que és possible integrar-la directament dins de l'aplicació mòbil, sembla més recomanable desplegar-la en un servidor extern al qual l'aplicació pugui connectar-se mitjançant una API REST. Aquest enfocament simplifica la integració i facilita considerablement l'escalabilitat del sistema.
- **Capacitat de generalització:** el model de xarxa neuronal ha de ser capaç de generalitzar adequadament quan s'aplica a dades noves, especialment aquelles capturades amb sensors diferents o en condicions variades respecte a les utilitzades durant l'entrenament inicial.

4.2.2 *Requisits de l'aplicació*

- **Multiplataforma:** l'aplicació mòbil ha de ser compatible amb diversos sistemes operatius (principalment *Android* i *iOS*).
- **Facilitat d'ús:** la interfície d'usuari haurà de ser intuïtiva i senzilla d'utilitzar, permetent als usuaris interactuar amb l'aplicació sense necessitat d'un procés d'aprenentatge complex.
- **Escalabilitat:** el disseny de l'aplicació ha de permetre fàcilment l'adaptació o ampliació de noves funcionalitats en futures actualitzacions, sense necessitat de grans esforços en la reestructuració del codi.
- **Integració de sensors:** per a la captura dels senyals d'ECG, l'aplicació utilitza els sensors de l'empresa *Movesense* [9], concretament els sensors del tipus *Sport*. L'avantatge principal d'aquests sensors és que disposen d'una API¹² oberta, la qual simplifica notablement la connexió i la interacció amb els diversos serveis que ofereix el sensor. *Movesense* també facilita diverses aplicacions senzilles d'exemple, les quals exemplifiquen clarament com establir la connexió amb el sensor i com fer ús de les seves funcionalitats bàsiques, ajudant així a accelerar el procés d'integració i desenvolupament de noves aplicacions.

¹² *Application Programming Interface*

5 Anàlisi dels Requisits Funcionals

5.1 Diagrama de Casos d'Ús

La següent *Figura* representa el Diagrama de Casos d'Ús de l'aplicació.

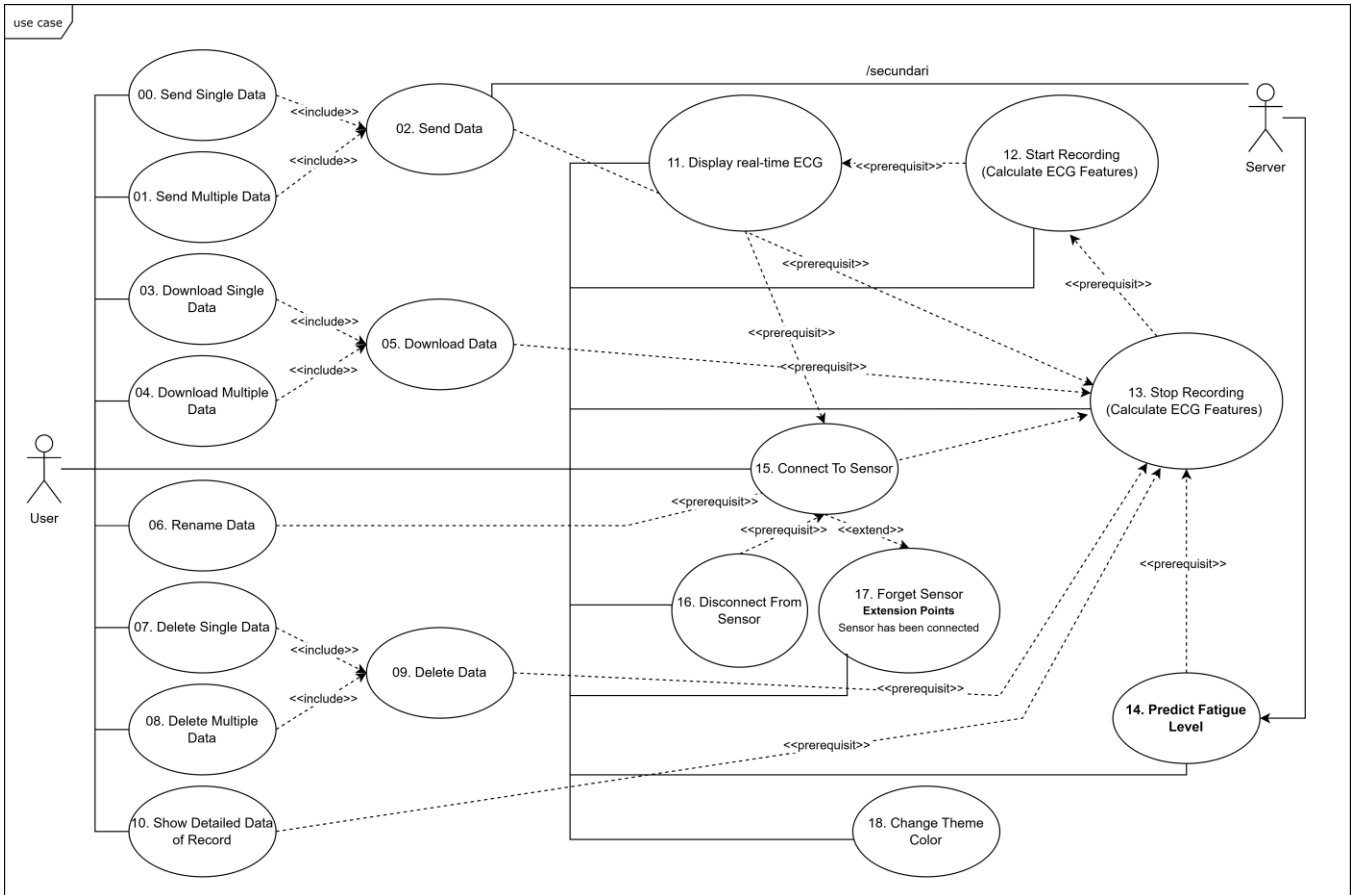


Figura 3. Diagrama de Casos d'Ús.

5.2 Diagrama de Classes

La següent *Figura* representa el Diagrama de Classes de l'aplicació.

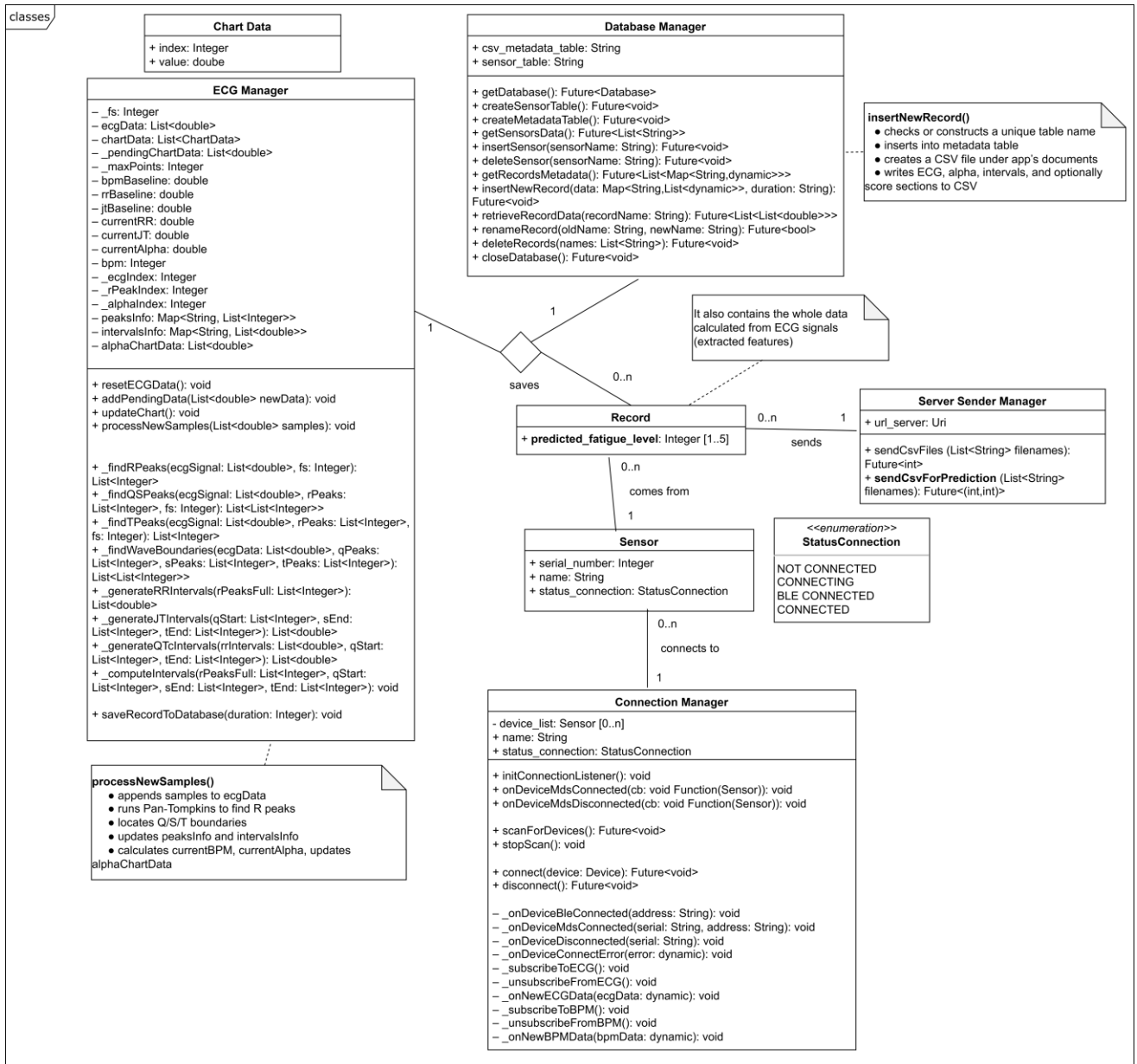


Figura 4. Diagrama de Classes.

6 Disseny

Per tal de poder avaluar el nivell de fatiga a partir de senyals d'ECG, la solució proposada ha consistit en el desenvolupament d'una CNN, inspirada en l'arquitectura VGG¹³. El sistema es basa en la conversió de senyals temporals d'ECG en representacions espectrogràfiques (imatges 2D) mitjançant la STFT¹⁴, que posteriorment s'utilitzen com a dades d'entrada del model.

6.1 Precedents

Tot i que l'objectiu final del projecte sempre ha sigut el desenvolupament d'un sistema intel·ligent capaç de detectar la fatiga, la idea inicial se centrava en la creació d'una aplicació que monitoritzés la fatiga en temps real a partir de l'extracció de característiques del senyal d'ECG. Tanmateix, això no resulta tant precís com la predicció utilitzant un model predictiu robust. Aquesta aplicació també havia de recollir i emmagatzemar les dades per poder-les analitzar posteriorment.

D'aquesta manera, el plantejament original consistia a utilitzar senyals d'ECG de dones amb càncer com a base per entrenar un model predictiu capaç d'estimar el seu nivell de fatiga. Tanmateix, no es disposava de cap criteri objectiu ni fiable per quantificar la fatiga, ni a partir de les dades fisiològiques ni mitjançant qüestionaris a les pacients.

Així doncs, el projecte va redirigir el focus cap a la cerca de bases de dades amb senyals ja analitzats i etiquetats segons nivells de fatiga. Això va permetre entrenar un model amb dades fiables, amb l'objectiu final d'aplicar-lo tant a dones amb càncer com a un grup de ciclistes col·laboradors d'ISGlobal, garantint així que el model pogués generalitzar-se i ser útil en diferents contextos.

6.2 Dades Inicials

Per desenvolupar una CNN robusta i fiable, era imprescindible trobar estudis que proporcionessin senyals d'ECG ja avaluats i classificats segons diferents nivells de fatiga, seguint criteris científics rigorosos. Un dels estudis més rellevants en aquest àmbit és el de Kazemnejad, A., Gordany, P. i Sameni, R., publicat el 2021 [10].

En aquest estudi, els autors van recollir dades de dos tipus de senyals fisiològics: senyals d'ECG i de FCG¹⁵, que és un registre dels sons i murmurs que fa el cor [11]. Van dur a terme una sèrie de proves d'estrès de 30 minuts amb 24 homes adults sans, sota condicions físiques diferents: repòs, caminada, cursa i bicicleta. Durant cada test, es van enregistrar els senyals d'ECG i FCG, i posteriorment es van etiquetar amb un nivell de fatiga assignat segons l'activitat realitzada.

La classificació de la **fatiga** es va fer en una **escala de l'1 al 6**, on:

- 1 indicava absència total de fatiga (repòs),
- 6 reflectia un alt nivell de fatiga (després d'un exercici intens com córrer).

¹³ *Visual Geometry Group*

¹⁴ *Short-Time Fourier Transform*

¹⁵ Fonocardiograma

Tots aquests registres van ser recopilats en una base de dades pública, amb accés lliure als fitxers de senyals d'ECG i una fulla de metadades que permet identificar fàcilment cada enregistrament. Els fitxers contenen primer una capçalera amb informació bàsica, seguida d'una llarga llista de valors numèrics que representen, punt per punt, la forma del senyal d'ECG.

6.3 Processament del Senyal d'Entrada

En una primera aproximació, el senyal d'ECG en brut es segmentava i es convertia directament en espectrogrames mitjançant la **STFT**. Aquesta estratègia va oferir resultats prometedors durant les fases d'entrenament i test, amb precisions que superaven el 96% i el 93%, respectivament. Tanmateix, quan es va posar a prova el model amb dades provinents de fonts diferents a les utilitzades per entrenar-lo, els resultats van ser decebedors: les prediccions dels nivells de fatiga no es corresponien gens amb la realitat de les mostres. El model, simplement, no era capaç de generalitzar.

Davant d'aquest repte, es va replantejar tot el procés de processat de les dades amb un objectiu clar: fer que el model fos realment robust i capaç de mantenir una bona capacitat predictiva, fins i tot amb senyals obtingudes en contextos i condicions diferents de les inicials.

6.3.1 Filtratge del Senyal

Un cop observats els mals resultats de generalització, es va sospitar que el problema principal era no haver filtrat prèviament el senyal d'ECG. Aquestes dades, recollides mitjançant sensors durant l'exercici físic, solen estar contaminades per diversos tipus de soroll. Aquest soroll pot distorsionar la forma del senyal i, com a conseqüència, dificultar enormement la capacitat del model per fer prediccions fiables fora del conjunt de dades d'entrenament [12].

Els tipus de soroll més habituals en senyals d'ECG enregistrats en moviment inclouen [12]:

- **Deriva de la línia base:** oscil·lacions lentes de baixa freqüència (0.1 – 0.3 Hz) causades per la respiració, moviments corporals o canvis en la impedància entre l'elèctrode i la pell.
- **Soroll electromiogràfic:** interferències generades per l'activitat muscular, que poden aparèixer en un rang ampli de freqüències (entre 5 i 500 Hz).
- **Interferència de la línia elèctrica:** soroll a 50 o 60 Hz procedent de fonts elèctriques properes, que pot distorsionar la forma dels pics del senyal d'ECG.
- **Soroll compost:** combinació dels diversos tipus de sorolls, que dificulten l'anàlisi precisa del senyal.

Per resoldre aquest problema, abans de convertir el senyal en espectrogrames, es va aplicar un procés de neteja. Es va utilitzar un filtre passa-alt per eliminar les freqüències més baixes, com les responsables de la deriva de la línia base. A més, es va usar un segon filtre específic per eliminar la interferència elèctrica, garantint així que el senyal d'entrada fos molt més clar i usable.

6.3.2 Normalització

Per aconseguir un model realment capaç de generalitzar, no n'hi havia prou amb filtrar el soroll: calia també garantir que totes les dades d'entrada estiguessin en la mateixa escala. Per això, es va aplicar un procés de normalització a cada senyal d'ECG.

Aquest pas consisteix a ajustar els valors del senyal tal com es mostra a la *Fórmula 1*. Això permet que el model no es vegi influenciat per diferències d'amplitud entre registres —que poden provenir de factors com el tipus de sensor, la intensitat de l'esforç o les condicions ambientals— i es pugui centrar exclusivament en la forma del senyal, que és el que realment importa per predir la fatiga.

$$x_{norm} = \frac{x - \mu}{\sigma} \quad (1)$$

6.3.3 Segmentació del Senyal i Remostreig

Els senyals d'ECG utilitzats com a entrada contien dades enregistrades durant intervals de 30 minuts, generant fitxers de gran mida i complexitat. Per tal de facilitar l'aprenentatge del model i optimitzar-ne el rendiment, es va decidir segmentar cada senyal en fragments de 1,000 punts. Aquesta estratègia no només reduïa la mida de cada mostra, sinó que també permetia augmentar el nombre total de segments disponibles per entrenar el model, afavorint així una millor generalització.

A més, cal tenir en compte que els senyals originals estaven enregistrats a una freqüència de mostreig de 8,000 Hz, molt superior a la necessària per representar adequadament la forma d'un senyal d'ECG. Per aquest motiu, es va aplicar un procés de remostreig (el que es coneix en anglès com *downsampling*) per reduir la freqüència a **200 Hz**, mantenint la informació essencial del senyal i eliminant redundàncies. Aquesta reducció permet conservar la informació rellevant del senyal ECG eliminant dades redundants, cosa que facilita un processament més eficient. A més, aquest ajust garanteix la coherència amb les mostres emprades per validar la capacitat de generalització del model, ja que aquestes també van ser enregistrades a 200 Hz.

6.3.4 Transformació a Espectrograms mitjançant STFT

La **STFT** és una eina fonamental en el processament de senyals, especialment eficaç per analitzar senyals no estacionaris com els senyals d'ECG. Permet examinar com evoluciona el contingut en freqüència d'un senyal al llarg del temps, cosa que és crucial per entendre i diagnosticar diverses condicions cardíques [13].

La **STFT** opera segmentant un senyal en el domini temporal en **intervals curts i solapats** mitjançant una funció finestra de longitud fixa. Cada segment es transforma després al domini de la freqüència mitjançant la transformada de Fourier. Aquest procés dona com a resultat una **representació bidimensional del senyal**, mostrant tant la informació temporal com la freqüencial [14].

Matemàticament, la STFT es descriu com:

$$STFT_f(t, \omega) = \int_{-\infty}^{+\infty} f(t) g(t - \tau) e^{-j\omega\tau} d\tau \quad (2)$$

On:

- $f(t)$ és el senyal original (un dels segments del senyal original).
- $g(t - \tau)$ és la finestra centrada en el temps τ .
- ω és la freqüència angular, és a dir, la freqüència que s'està mirant dins d'una finestra temporal concreta. Es relaciona amb la freqüència f com mostra la *Fórmula 3*.

$$\omega = 2\pi f \quad (3)$$

- $e^{-j\omega t}$ és una sinusoide complexa utilitzada per descompondre el senyal en les seves components de freqüència.

Els senyals d'ECG són inherentment no estacionaris, és a dir, el seu contingut en freqüència varia al llarg del temps a causa de factors fisiològics i patològics [13]. La STFT és especialment adequada per analitzar aquests senyals perquè proporciona una representació temps-freqüència, permetent observar esdeveniments transitoris i canvis dinàmics en l'activitat elèctrica del cor [14].

En el cas d'aquest projecte, cada segment del senyal d'ECG contenia 1,000 punts. Aquests segments es van transformar en espectrogrames 2D utilitzant la STFT amb els següents paràmetres:

- Mida de la finestra: 256 punts.
- Salt entre finestres: 128 punts (solapament parcial del 50%).

El nombre de finestres que s'obtenen d'un segment es calcula com:

$$\text{nombre finestres} = \left\lfloor \frac{\text{mida segment} - \text{mida finestra}}{\text{mida salt}} \right\rfloor + 1 \quad (4)$$

Substituint:

$$\text{nombre finestres} = \left\lfloor \frac{1,000 - 256}{128} \right\rfloor + 1 = \lfloor 5.81 \rfloor + 1 = 5 + 1 = 6$$

És a dir, s'obtenen 6 finestres temporals per segment.

D'altra banda, quan s'aplica la transformada de Fourier a un senyal real d'ECG, l'espectre de freqüència és simètric respecte de la freqüència 0. Això vol dir que només es necessita la meitat més un dels valors per tenir tota la informació útil:

$$F(-\omega) = \overline{F(\omega)} \quad (5)$$

Així doncs, el nombre de valors de freqüència (o *bins*) que es conserven per cada finestra és:

$$\text{valors de freqüència} = \frac{\text{mida finestra}}{2} + 1 \quad (6)$$

Substituint:

$$\text{valors de freqüència} = \frac{256}{2} + 1 = 129$$

El resultat és un **espectrograma** de mida:

$$\text{Espectrograma} = (6 \text{ finestres temporals}, 129 \text{ bins de freqüència})$$

Cada cel·la d'aquesta matriu representa la intensitat logarítmica d'una freqüència específica en un instant concret. Això permet visualitzar clarament l'evolució temporal del contingut en freqüència del segment del senyal d'ECG. La *Figura 5* mostra un espectrograma (transposat) per cada nivell de fatiga, pintats en funció de la intensitat freqüencial.

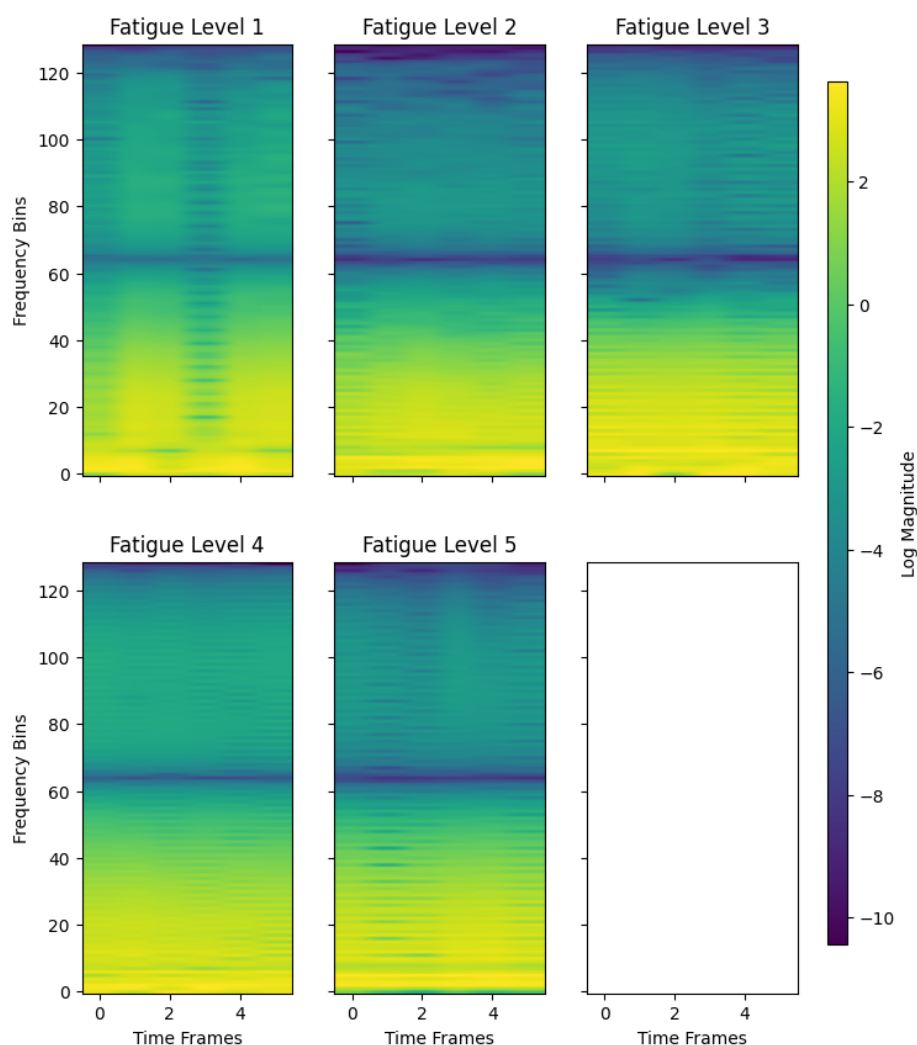


Figura 5. Espectrograms obtinguts després del processament dels senyals d'ECG.

Aquests espectrograms 2D van ser les dades d'entrada del model.

6.4 Arquitectura del Model

Per resoldre el problema predictiu plantejat en aquest projecte, es va optar per construir una CNN basada en l'arquitectura VGG, una de les arquitectures més icòniques i efectives en l'àmbit de la visió per computador.

La VGG és una arquitectura de CNN desenvolupada per Karen Simonyan i Andrew Zisserman de la Universitat d'Oxford. Aquesta xarxa va assolir una gran notorietat després de destacar en el repte *ImageNet Large Scale Visual Recognition Challenge* l'any 2014, aconseguint una precisió impressionant del 92,7% [15]. La clau de la seva eficàcia radica en

una estructura senzilla però profunda, amb blocs repetits de **convolucions 3×3** i capes de *max-pooling* per reduir la dimensionalitat de les imatges [15].

A diferència de models anteriors que feien servir filtres, o nuclis de convolució, de mida més gran (com 11×11 o 5×5), VGG va apostar per molts nuclis de convolució petits (3×3) apilats en profunditat, cosa que permet extreure característiques de manera més precisa i mantenir el control sobre el nombre de paràmetres [15].

L'arquitectura VGG16 (veure *Figura 6*), una de les més utilitzades, es compon de [15]:

- 13 capes convolucionals (agrupades en 5 blocs),
- 5 capes de *max-pooling* (una després de cada bloc),
- 3 capes totalment connectades (el que es coneix com *fully connected*) al final, dues de 4,096 neurones i una de classificació amb activació *SoftMax*.

Aquest enfocament modular i profund ha fet que la VGG sigui àmpliament adoptada en *transfer learning* [15] —és una tècnica de ML que consisteix a aprofitar el coneixement adquirit per un model entrenat en una tasca prèvia per aplicar-lo a una nova tasca relacionada. Això permet reduir el temps d'entrenament i millorar el rendiment, especialment quan es disposa de poca informació per a la nova tasca [16] — i adaptacions personalitzades, com en aquest projecte.

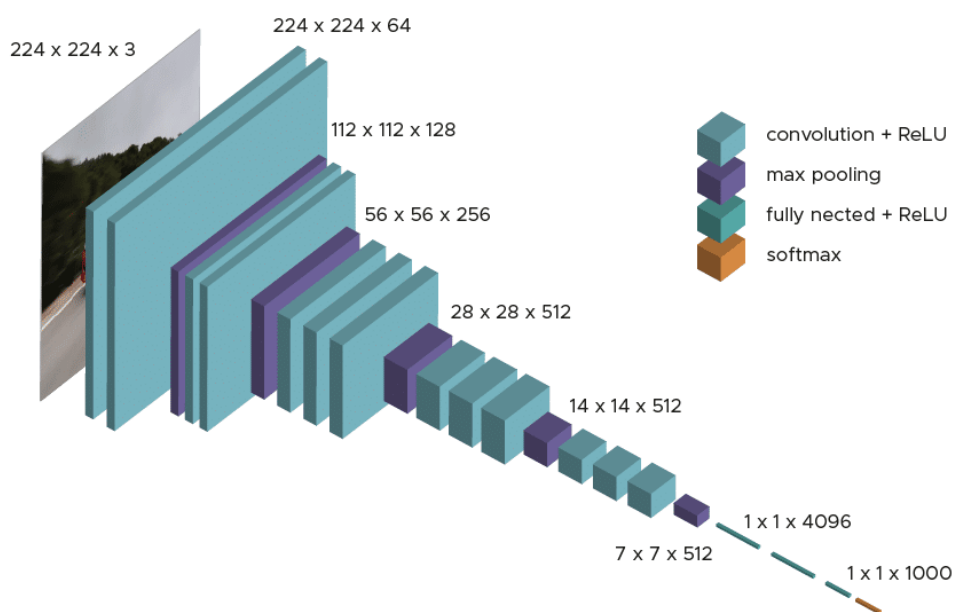


Figura 6. Arquitectura del model VGG16.

6.5 Adaptació del Model VGG

Partint de l'arquitectura clàssica de VGG16, es va dissenyar una variant adaptada a les característiques específiques de les dades del projecte. El model original VGG16 va ser concebut per a tasques de classificació d'imatges i espera com a entrada imatges de mida 224×224 píxels amb 3 canals (RGB¹⁶), corresponents als colors vermell, verd i blau. En canvi, en aquest cas, el model rep com a entrada espectrogrames. El color no aporta

¹⁶ Red Green Blue

informació rellevant, de manera que les imatges d'entrada tenen només 1 canal (escala de grisos) i una mida de 6×129 .

A més, mentre que el model VGG16 originalment es va pensar per classificar 1,000 categories d'imatge, el model d'aquest projecte està pensat per **distingir entre 5 nivells de fatiga**¹⁷.

El model construït es pot resumir en les següents fases:

6.5.1 Fase Convolucional (Extracció de Característiques)

La xarxa està formada per **5 blocs convolucional**s:

- Cada bloc conté entre 2 i 3 capes convolucional, com es pot veure en la *Figura 6*. Totes aquestes capes utilitzen **nuclis de convolució de mida 3×3** i la funció d'activació **ReLU**, que permet introduir no-linealitat i millorar la capacitat del model per aprendre patrons complexos. Aquesta estructura segueix la mateixa filosofia que l'arquitectura original de VGG16.
- Després de cada bloc convolucional, s'aplica una capa de **max-pooling** amb una **finestra de 1×2** en lloc de la típica 2×2 . Aquesta operació serveix per reduir la mida de les dades i, al mateix temps, conservar la informació més rellevant. Funciona escollint el valor màxim dins d'una petita regió i utilitzant-lo com a representació d'aquesta zona.
En aquest model, la finestra 1×2 significa que només es redueixen les columnes, que corresponen a les freqüències de l'espectrograma. En canvi, les files —que representen el temps— no es redueixen. Això és així perquè l'eix temporal només té 6 valors (6 finestres temporals), i reduir-lo podria fer que el model perdés informació important sobre com evoluciona el senyal en el temps.

Els blocs s'organitzen amb nombre de filtres creixent: $64 \rightarrow 128 \rightarrow 256 \rightarrow 512 \rightarrow 512$, per tal de capturar progressivament característiques més abstractes i complexes.

6.5.2 Fase d'Aplanament

Un cop finalitzada l'extracció de característiques, s'obté com a resultat una matriu tridimensional: una espècie de cub amb alçada, amplada i profunditat (aquesta última correspon als filtres dels blocs convolucional). La *Figura 7* mostra l'evolució de la forma dels espectrograms a mida que avancen pels diferents blocs convolucional.

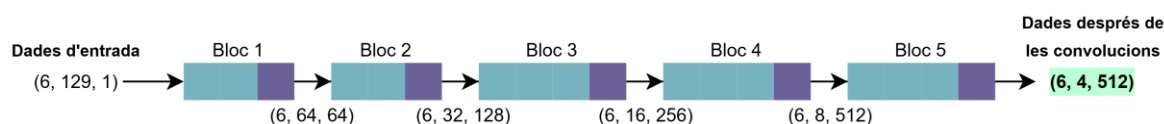


Figura 7. Evolució de la forma dels espectrograms en les etapes de convolució.

Tanmateix, les capes *fully connected* que venen després de les convolucional no poden treballar directament amb dades en forma de matriu tridimensional (com les que

¹⁷ Veure apartat “*Redefinició dels Nivells de Fatiga de les Dades Inicials*” per entendre el motiu de predir 5 nivells de fatiga en lloc dels 6 inicials.

surten de les convolucions). Aquestes capes funcionen com una xarxa de neurones clàssica, on cada neurona està connectada a totes les sortides de la capa anterior.

Perquè això sigui possible, cal reorganitzar la informació, és a dir, cal transformar aquesta estructura tridimensional en un vector unidimensional. Aquesta transformació s'anomena aplanament i es fa just abans d'entrar a la primera capa *fully connected*.

6.5.3 Fase de Classificació

Les últimes tres capes de la CNN són **capes *fully connected***, responsables de fer la classificació final un cop s'han extret totes les característiques rellevants.

- Les dues primeres capes *fully connected* contenen **2,048 neurones** cadascuna i utilitzen la funció d'activació **ReLU**. Per evitar que el model s'adapti massa als exemples d'entrenament (**overfitting**), s'aplica una tècnica anomenada **Dropout**, que consisteix a desactivar aleatòriament el 50% de les connexions durant l'entrenament. Això obliga el model a aprendre patrons més generals i robustos.
- L'última capa és una capa *fully connected* amb **5 neurones**, una per a cada **nivell de fatiga** que es vol predir. Aquesta capa utilitza l'activació **SoftMax**, que transforma les sortides en una distribució de probabilitats: cada valor representa la probabilitat que l'entrada correspongui a una de les 5 classes.

Com que un senyal d'ECG es divideix en diversos segments per facilitar-ne l'anàlisi, cada fragment es transforma en un espectrograma i es classifica de manera independent. El model genera, per tant, una **predicció (una distribució de probabilitats)** per a cada segment.

Per tal d'obtenir una predicció global per al senyal complet, es recull la classe amb la probabilitat més alta de cada segment. A continuació, es determina quina d'aquestes classes apareix amb més freqüència i aquesta es considera com la **classe final** predita pel model. A més, es pot calcular la probabilitat mitjana d'aquesta classe majoritària entre tots els segments en què ha estat seleccionada, oferint així una **mesura de confiança en la predicció final**.

6.6 Redefinició dels Nivells de Fatiga de les Dades Inicials

Inicialment, els senyals d'ECG estaven classificats en **6 nivells diferents de fatiga**, i es va entrenar el model utilitzant aquestes 6 classes.

Tot i que els resultats obtinguts durant les fases d'entrenament, validació i test van ser bons, el model presentava un problema important: **no era capaç de generalitzar bé**. Això vol dir que, quan es feien prediccions sobre mostres completament noves (que s'havien pres sobre altres individus, amb sensors i freqüències de mostreig diferents) les classificacions que retornava no coincidien amb la realitat del nivell de fatiga de la persona.

La causa principal d'aquest problema era un fort **desbalanceig de classes**. Cinc dels nivells de fatiga disposaven d'un nombre similar de mostres, mentre que la **classe 2** (corresponent a un estat de repòs assegut en una cadira) només tenia **3 mostres**. Concretament, la **classe 1** disposava de **8 mostres**, les **classes 3 i 4** de **13 mostres**, la **classe 5** de **10 mostres**, i la **classe 6** d'**11 mostres**.

Aquest desequilibri va afectar negativament el rendiment del model perquè, durant l'entrenament, el model va veure moltes més mostres d'unes classes que d'altres. Com a conseqüència, va aprendre a reconèixer molt millor les classes més freqüents, i tendia a ignorar o malclassificar les menys representades. Això va provocar que, tot i que pogués tenir bons resultats sobre les dades vistes, **no era fiable** quan havia de fer prediccions en condicions noves.

Per aquest motiu, i tenint en compte que el **primer nivell de fatiga ja representava un estat de repòs**, es va decidir **eliminar la classe 2** per reduir el desequilibri i simplificar el problema.

Així, es va redefinir la classificació en **5 nivells de fatiga**, que són els següents:

1. **Nivell 1 de fatiga**: senyal d'ECG en estat de repòs estirat i relaxat.
2. **Nivell 2 de fatiga**: els subjectes caminen a una velocitat constant de 3,7 km/h (activitat lleugera).
3. **Nivell 3 de fatiga**: els subjectes pedalen en una bicicleta estàtica a una velocitat constant.
4. **Nivell 4 de fatiga**: els subjectes pedalen en bicicleta estàtica, però augmenten la càrrega fins a sentir fatiga.
5. **Nivell 5 de fatiga**: els subjectes corren amb una intensitat progressiva fins arribar a l'esgotament.

6.7 Definició dels Conjunts d'Entrenament, Validació i Test

Un cop redefinides les classes de fatiga i generats els espectrogrames a partir dels senyals d'ECG segmentats, es va procedir a l'entrenament del model final.

Per assegurar una avaluació justa del rendiment del model, es va reservar un **10% dels espectrogrames com a conjunt de test final**. Aquest conjunt no es fa servir en cap moment durant l'entrenament i va servir per mesurar la capacitat del model de predir correctament en dades que mai ha vist, tot i pertànyer a la mateixa cohort.

Els **90% restants** dels espectrogrames es van utilitzar per a **entrenar** el model. Tot i tenir l'arquitectura definida, quedaven per ajustar diversos **hiperparàmetres** —valors que no apren el model, però que influeixen molt en com s'entrena. Entre aquests, hi destaquen:

- **LR**¹⁸: determina la velocitat amb què el model actualitza els seus pesos a cada pas. Un valor massa alt pot fer que no aprengui bé, i un de massa baix pot fer que aprengui molt lentament.
- **BS**¹⁹: és la quantitat d'exemples que el model processa abans d'actualitzar els pesos. Per exemple, un BS de 32 vol dir que el model fa una predicció sobre 32 espectrogrames, calcula l'error i ajusta els pesos.

¹⁸ *Learning Rate*

¹⁹ *Batch Size*

- **ES**²⁰: és una tècnica per evitar que el model s'entreni massa i aparegui l'*overfitting*. Si després de diversos intents l'error en el conjunt de validació no millora, s'atura automàticament l'entrenament [17].
- **LRS**²¹: és una estratègia per anar reduint el LR a mesura que avança l'entrenament. Això ajuda a fer ajustos més precisos al final del procés, quan ja està prop d'una solució òptima [18].

Per trobar la millor combinació d'aquests hiperparàmetres, es va fer servir la **validació creuada en 3 parts**. Aquesta tècnica consisteix a dividir el conjunt d'entrenament en tres parts iguals. A cada iteració:

- dues parts s'utilitzen per entrenar el model,
- i la tercera per validar-lo.

Això es repeteix tres cops, intercanviant les parts, de manera que cada subgrup fa de conjunt de validació una vegada. Aquesta estratègia ajuda a obtenir una estimació més robusta del rendiment del model i a triar la configuració que ofereix els millors resultats de manera general. La *Figura 8*²² mostra els resultats obtinguts després d'aplicar aquesta estratègia.

²⁰ *Early Stopping*

²¹ *Learning Rate Scheduler*

²² No es van arribar a provar totes les combinacions que utilitzaven un LR de 0.001 perquè els resultats que es van anar obtenint amb aquest eren clarament molt deficients.

Training and Validation Metrics per Epoch for each Model

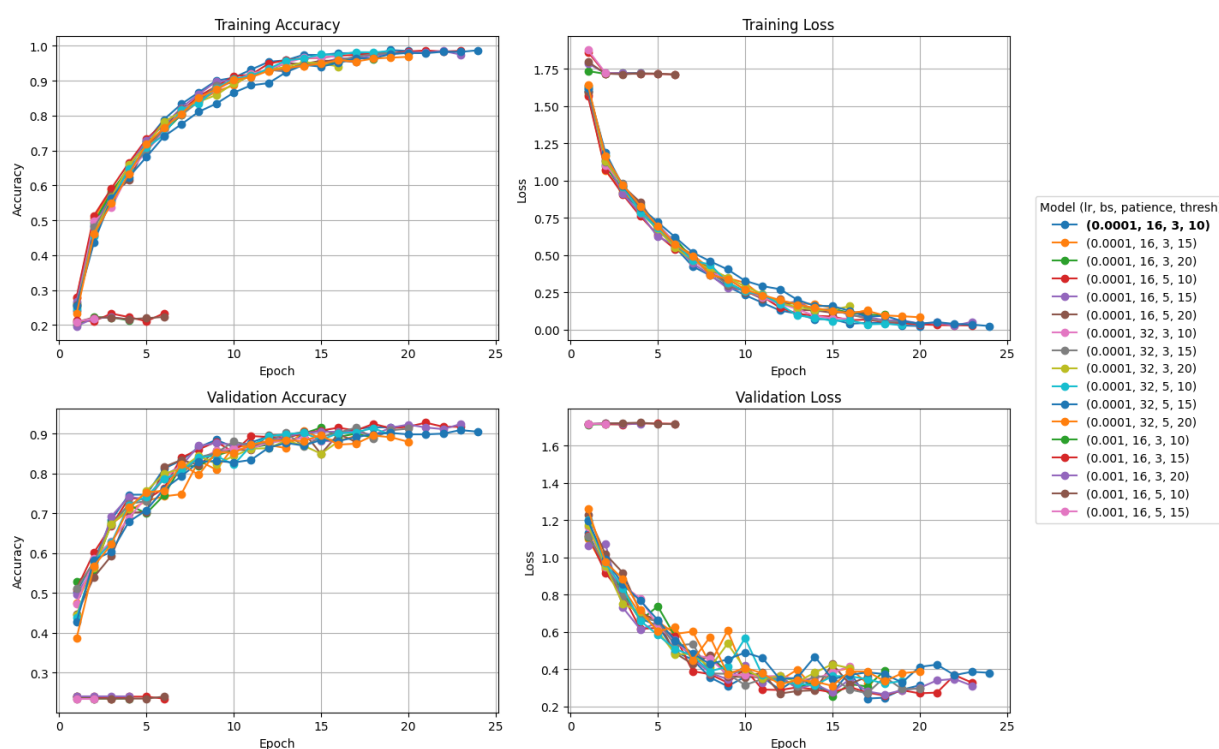


Figura 8. Evolució de les mètriques d'entrenament i validació segons la combinació d'hiperparàmetres.

Un cop identificada la millor combinació d'hiperparàmetres (ressaltada en negreta a la *Figura 8*), es van fixar els següents valors per entrenar el model final:

- LR de 0.0001.
- BS de 16.
- ES amb una paciència de 3 èpoques (si el model no millora durant 3 èpoques consecutives, s'atura l'entrenament).
- LRS on a partir de la desena època, la velocitat d'aprenentatge comença a reduir-se per afinar millor l'ajust.

Amb aquests paràmetres, es va entrenar el model final utilitzant tot el conjunt d'entrenament (el 90% de les dades), per aprofitar al màxim la informació disponible.

6.8 Disseny de l'Aplicació Mòbil

Inicialment es va començar a desenvolupar l'aplicació mòbil, però aquest desenvolupament es va aturar temporalment per tal de prioritzar la creació del model predictiu. Un cop desenvolupat el model, s'ha aprofitat per integrar-lo directament a l'aplicació.

L'aplicació es va pensar per mostrar 3 pantalles diferents:

1. Pantalla *Home*

- Monitoratge en temps real:** mostra contínuament el senyal d'ECG en directe, oferint una visualització clara i immediata.

- b. **Característiques instantànies:** paral·lelament al senyal, es mostren paràmetres importants derivats directament de l'ECG (representats a la *Figura 1*), com ara intervals R-R.
- c. **DFA- $\alpha 1$:** mostra en directe l'evolució d'aquest paràmetre, que permet analitzar la complexitat i autocorrelació dels pics R [19]:
 - i. Valors propers a **1** indiquen un patró de variabilitat altament correlacionat, característic d'esforços lleus o condicions de repòs.
 - ii. Valors al voltant de **0,75** indiquen el llindar aeròbic, on l'organisme comença a generar lactat de forma constant però estable.
 - iii. Valors propers a **0,5 o inferiors** indiquen una pèrdua de correlació i una variabilitat més aleatòria, associada a exercici intens o superació del llindar anaeròbic.

2. Pantalla *Records*

- a. **Historial d'enregistraments:** permet accedir fàcilment a totes les sessions d'ECG gravades, que s'emmagatzemen localment a l'aplicació.
- b. **Visualització detallada:** en seleccionar qualsevol gravació, l'usuari pot revisar el traçat complet de l'ECG enregistrat, juntament amb les característiques calculades.
- c. **Predicció del nivell de fatiga:** aquí és on intervé la **integració amb el model construït**. Degut a possibles limitacions de connectivitat, la predicció del nivell de fatiga no és automàtica. En canvi, l'usuari disposa d'un botó específic **Predict Fatigue Level** dins de cada registre, que, en prémer-lo, envia les dades guardades del senyal d'ECG a un servidor local que retorna immediatament una estimació precisa del nivell de fatiga.

3. Pantalla *Profile*

- a. **Informació tècnica completa:** aquesta pantalla proporciona informació específica sobre el sensor connectat, incloent model, estat de bateria, versió de *firmware* i altra informació tècnica rellevant.
- b. **Accés directe a suport extern:** també incorpora un enllaç directe a la pàgina web oficial del fabricant, facilitant als usuaris la consulta ràpida i senzilla d'informació addicional, guies tècniques o suport específic.

7 Implementació

7.1 Entorn de Desenvolupament

El desenvolupament del projecte s'ha dut a terme utilitzant una *Jupyter Notebook*.

- **Llenguatge de programació:** *Python* (versió 3.11.4 [20]).
- **Llibreries principals utilitzades:**
 - **Manipulació i visualització de dades:**
 - *numpy* (versió 2.1.3 [21])
 - *matplotlib* (versió 3.9.2 [22])
 - *seaborn* (versió 0.13.2 [23])
 - *pandas* (versió 2.2.3 [24])
 - **Processament del senyal d'ECG:**
 - *neurokit2* (versió 0.2.10 [25])
 - **Modelatge de la xarxa neuronal:**
 - *tensorflow.keras* (versió 3.8.0 [26]): per definir, entrenar i avaluar la CNN.
 - **Avaluació del rendiment:**
 - *scikit-learn* (versió 1.5.2 [27]): utilitzada per calcular mètriques com l'exactitud, la precisió, el *recall* o la matriu de confusió.

7.2 Processament del Senyal

7.2.1 Filtratge del Senyal

Per tal de netejar el senyal d'ECG, es va utilitzar la funció *ecg_clean* [28] de la llibreria *neurokit2*. Aquesta funció, a través del mètode *neurokit*, aplica principalment dos passos de filtratge seguits per millorar la qualitat del senyal:

1. Filtre passa-alt *Butterworth*

Aquest filtre s'encarrega principalment d'atenuar i/o eliminar la deriva de la línia base.

- **Tipus de filtre:** és un filtre *Butterworth*, conegut per tenir una **resposta en freqüència suau**, sense oscil·lacions en la banda de pas (on el senyal es manté) i una **transició suau però eficient** cap a la banda de rebuig (on el senyal s'atenua) [29]. Això ajuda a conservar la forma original del senyal d'ECG.
- **Freqüència de tall:** s'estableix a **0.5 Hz**, de manera que es filtren les freqüències per sota d'aquest valor, que són considerades soroll de baixa freqüència.
- **Ordre del filtre:** és un filtre d'**ordre 5**. L'ordre del filtre determina com de ràpid és el canvi entre la banda de pas i la banda de rebuig. Un ordre 5 és un bon compromís entre netejar bé la deriva [30] i no deformar les components útils (com els pics R) que comencen a partir de 1-2 Hz [12].

2. Filtre de línia elèctrica

Aquest filtre s'utilitza per eliminar el soroll de la xarxa elèctrica que sol aparèixer a 50 Hz (o 60 Hz en altres països).

- **Tipus de filtre:** Es fa servir un filtre *notch*, també conegut com a filtre **rebuig de banda**. Aquest filtre està pensat per eliminar només una franja molt estreta de freqüències, sense afectar la resta del senyal.
- **Freqüència objectiu:** En aquest cas, el filtre està **centrat a 50 Hz** — que és la freqüència amb la que opera la xarxa elèctrica espanyola —, eliminant les components del senyal que oscil·len just al voltant d'aquesta freqüència (aproximadament entre **49 Hz i 51 Hz**).
- **Paràmetres:** S'indica a la funció amb *method="powerline"* i *powerline=50*, per especificar que es vol eliminar el soroll típic de la xarxa elèctrica europea.

7.3 Implementació de la CNN

La CNN es va construir utilitzant l'API *Sequential* de la llibreria *tensorflow.keras*, que permet afegir les capes de forma ordenada i intuïtiva.

7.3.1 Blocs convolucional

La xarxa està formada per cinc blocs convolucional. Cada bloc inclou diverses capes *Conv2D*, que s'encarreguen d'extreure característiques de l'espectrograma, seguides d'una capa de *MaxPooling2D* per reduir progressivament la dimensió de les dades.

A la **primera capa convolucional** es va definir el paràmetre *input_shape*, que especifica la forma de l'espectrograma d'entrada (alçada, amplada i canals). Aquesta informació era necessària perquè la xarxa pogués saber com eren les dades d'entrada, però només cal indicar-la a la primera capa; les següents la dedueixen automàticament a partir de la sortida de la capa anterior.

D'altra banda, per reduir la mida de l'espectrograma tal com s'ha comentat a l'apartat de *Disseny*, a les capes de *max-pooling* es van definir els paràmetres següents:

- *kernel_size=(1, 2)*: indica que la finestra que s'aplica per seleccionar valors té una alçada d'1 píxel i una amplada de 2. Això significa que el filtrat es fa sobre parelles de columnes, mantenint intacta l'alçada.
- *strides=(1, 2)*: controla de quant es mou la finestra en cada pas. En aquest cas, es desplaça 1 posició en vertical i 2 en horitzontal, cosa que redueix l'amplada de l'espectrograma però manté la seva alçada.

7.3.2 Blocs fully connected

Un cop finalitzada la part convolucional, es va fer servir la capa *Flatten*, que transforma la sortida multidimensional en un vector unidimensional, preparat per ser processat per les capes densament connectades.

A continuació, es van afegir dues capes de tipus *Dense (fully connected)*. Entre aquestes es van incloure dues capes de tipus *Dropout (0.5)* per tal de reduir l'*overfitting*, com s'ha comentat en l'apartat de *Disseny*.

Finalment, es va afegir una darrera capa de tipus *Dense* amb **5 neurones** i activació *SoftMax*, que permet obtenir una probabilitat per a cadascun dels **5 nivells de fatiga** que es volen predir.

7.4 Entrenament del Model

Per dividir els conjunts d'entrenament i test, es va utilitzar la funció *train_test_split* de la llibreria de *scikit-learn*. Seguidament, per optimitzar el rendiment del model i seleccionar la millor combinació d'hiperparàmetres, es va utilitzar la funció *KFold(n_splits=3)*, també de *scikit-learn*, per tal d'aplicar la validació creuada en 3 parts.

En cada *fold* de la validació creuada, el model es compilava especificant l'optimitzador, la funció de pèrdua i les mètriques que es volien monitoritzar:

- ***optimizer=Adam(learning_rate=lr)***: s'utilitza l'optimitzador *Adam*, molt habitual en DL pel seu bon rendiment i adaptació dinàmica del LR. El valor de LR s'extreu d'un conjunt predefinit de valors (0.0001 i 0.001) per fer recerca d'hiperparàmetres.
- ***loss='sparse_categorical_crossentropy'***: funció de pèrdua (criteri que el model intenta minimitzar durant l'entrenament) per a classificació multiclasse amb etiquetes enteres (0 a 4). Aquesta funció compara la distribució de probabilitats predita pel model amb l'etiqueta real i calcula una penalització en funció de la distància entre ambdues. El terme *sparse* fa referència al fet que no cal codificar les etiquetes amb *one-hot encoding*, sinó que poden ser números enters directament.

Matemàticament, la *loss* es calcula com:

$$loss = -\frac{1}{N} \sum_{i=1}^N \log(\hat{y}_i) \quad (7)$$

on \hat{y}_i és la probabilitat assignada a la classe correcta per la mostra i , i N és el nombre d'espectrogrames.

- ***metrics=['accuracy']***: es monitoritza l'*accuracy* per fer seguiment del rendiment durant l'entrenament i la validació.

D'altra banda, també es van utilitzar *callbacks*, que són funcions que s'executen automàticament durant l'entrenament que permeten controlar el procés d'aprenentatge de manera flexible:

- **ES**: serveix per aturar l'entrenament de manera anticipada si la pèrdua de validació (*val_loss*) no millora després d'un nombre determinat d'èpoques (*patience*). Evita l'*overfitting* i redueix el temps computacional.
- **LRS**: permet ajustar dinàmicament el valor del LR durant l'entrenament. Evita canvis bruscos en les primeres èpoques i afavoreix ajustos més fins a mesura que l'entrenament avança. En aquest cas, es va usar la funció *scheduler*:
 - Si l'època actual (*epoch*) és menor que un valor llindar (*epoch_threshold*), es manté el LR constant.
 - A partir del llindar, la taxa d'aprenentatge comença a disminuir exponencialment, segons la fórmula:

$$LR_{nou} = LR_{actual} \times e^{-0.1} \quad (8)$$

Un cop entrenat el model en cada una de les 3 particions de la validació creuada, es disposava d'un conjunt d'històrics d'entrenament per a cada partició. Cada un d'aquests històrics contenia, per cada època, els valors de mètriques com la pèrdua (*loss*) i la precisió (*accuracy*) tant en entrenament com en validació.

No obstant això, la durada de l'entrenament pot variar entre particions a causa de l'ús del *callback* ES, que atura automàticament l'entrenament quan no es detecta millora. Això fa que cada *fold* tingui un nombre d'èpoques diferent.

Per calcular unes mètriques globals representatives del model, es va utilitzar la funció `compute_metrics_mean`. Aquesta funció:

1. Recull totes les mètriques obtingudes a cada *fold* (per exemple, *val_loss*, *val_accuracy*, etc.).
2. Omple les diferències de longitud entre *folds* amb valors *NaN* per tal d'unificar-los en una matriu de la mateixa mida.
3. Calcula la mitjana de cada mètrica per època, ignorant els valors *NaN*, per no penalitzar els models que han acabat abans per convergència.

Això va permetre obtenir, per cada combinació d'hiperparàmetres, un perfil mitjà del comportament del model (pèrdua i precisió) al llarg del temps, que és el que està representat en la *Figura 8*.

Un cop finalitzada la fase de validació, es va entrenar el model final utilitzant tot el conjunt d'entrenament (el 90% de les dades), aplicant els paràmetres amb els valors òptims obtinguts durant aquesta fase.

7.5 Implementació de l'Aplicació Mòbil

Per implementar l'aplicació s'ha utilitzat **Flutter** (versió 3.32.0 [31]), un *framework* desenvolupat per *Google* que permet crear aplicacions multiplataforma des d'un únic codi font. A més, *Movesense* disposa d'un *plugin* per *Flutter*, **mdsflutter** [32], que facilita la utilització de l'API per connectar-se i comunicar-se amb els seus sensors.

Per gestionar les dades de manera eficient, l'aplicació incorpora una **base de dades local SQLite**, que permet consultes ràpides i persistència de la informació sense necessitat de connexió externa. Aquesta base de dades conté dues taules principals:

- *movesense_timeinfo*: emmagatzema les metadades de cada enregistrament, com ara el nom assignat, la duració de la sessió o la data i hora de creació.
- *movesense_sensors*: guarda informació dels sensors, com el nom i el número de sèrie, facilitant la gestió de dispositius vinculats.

Tot i que la base de dades és essencial per a una gestió interna eficient, també s'utilitzen **fitxers CSV** per enregistrar les dades del senyal d'ECG i les característiques derivades. Aquest format és lleuger i ideal per a la transferència ràpida d'informació entre l'aplicació i un servidor local *Python*, que s'encarrega del processament.

El servidor, construït amb *Flask* (versió 3.0.3 [33]), rep les dades via REST API, processa la informació predint el nivell de fatiga, i el retorna a l'aplicació.

8 Avaluació

Després de completar l'entrenament final, el model es va validar emprant el 10% de les dades restants (**conjunt de test**), les quals no havien estat utilitzades durant l'entrenament. Aquesta validació va permetre mesurar el comportament del model amb dades no vistes, però provinents de la mateixa cohort.

Es va obtenir la següent matriu de confusió:

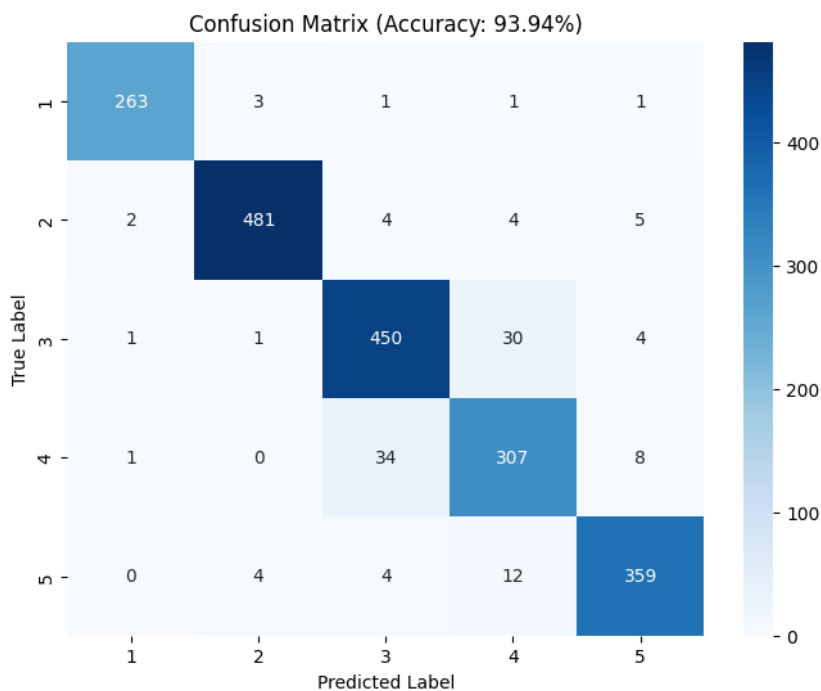


Figura 9. Matriu de confusió de les prediccions del model sobre el conjunt de test.

A partir de la matriu de confusió (veure *Figura 9*), es poden extreure diferents mètriques per avaluar el rendiment del model.

8.1 Mètriques de Rendiment Global

Precisió global (*accuracy*): és la proporció de prediccions correctes (tant positives com negatives) respecte al total. Es calcula com:

$$\text{Precisió global (accuracy)} = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

On:

- TP^{23} : Nombre de casos positius que el model ha classificat correctament com a positius.
- TN^{24} : Nombre de casos negatius que el model ha classificat correctament com a negatius.

²³ True Positives

²⁴ True Negatives

- FP²⁵: Nombre de casos negatius que el model ha classificat incorrectament com a positius.
- FN²⁶: Nombre de casos positius que el model ha classificat incorrectament com a negatius.

Substituint:

$$\text{Precisió global (accuracy)} = \frac{1860}{1980} = 0.9394 = 93.94\%$$

8.2 Mètriques per Classe

Per una anàlisi més precisa, es pot calcular el rendiment del model per cada classe individual mitjançant les mètriques següents:

- **Precisió**: proporció de prediccions positives que realment són positives. Es calcula com:

$$\text{Precisió} = \frac{TP}{TP + FP} \quad (10)$$

- **Sensibilitat (recall)**: proporció de positius reals que han estat correctament detectats. Es calcula com:

$$\text{Sensibilitat (recall)} = \frac{TP}{TP + FN} \quad (11)$$

- **F1 score**: mitja harmònica entre precisió i sensibilitat. Es calcula com:

$$\text{F1 score} = 2 \times \frac{\text{Precisió} \times \text{Sensibilitat}}{\text{Precisió} + \text{Sensibilitat}} \quad (12)$$

- **Especificitat**: proporció de negatius reals que han estat classificats correctament com a negatius. Es calcula com:

$$\text{Especificitat} = \frac{TN}{TN + FP} \quad (13)$$

- **Taxa d'error**: proporció de prediccions incorrectes. Es calcula com:

$$\text{Taxa d'error} = \frac{FP + FN}{TP + TN + FP + FN} \quad (14)$$

La Taula 1 mostra un resum d'aquestes mètriques per classe.

Classe	TP	FP	FN	TN ²⁷	Precisió	Recall	F1 score	Especificitat	Taxa d'error
1	263	4	6	1707	0.985	0.978	0.981	0.998	0.5%
2	481	8	15	1476	0.984	0.970	0.977	0.994	1.16%

²⁵ False Positives

²⁶ False Negatives

²⁷ Es pot calcular com la diferencia entre el nombre total de mostres i la suma de TP, FP i FN de la classe *i*.

3	450	43	36	1451	0.913	0.926	0.919	0.971	3.99%
4	307	47	43	1583	0.867	0.877	0.872	0.971	4.55%
5	359	18	20	1583	0.952	0.947	0.950	0.989	1.92%

Taula 1. Resum de les mètriques per classe.

D'altra banda, també es pot calcular l'**AUC**²⁸ (veure *Figura 10*), que és una mètrica que mesura la **capacitat del model per distingir correctament entre les diferents classes**, independentment del llindar de classificació que s'utilitzi [34]. Aquesta mètrica es basa en la corba ROC, que representa la relació entre la **sensibilitat** (*recall*) i la **taxa de FP** per a diferents valors de llindar. L'AUC pren valors entre 0 i 1, on un valor de 1.0 indica una separació perfecta entre classes, i un valor de 0.5 indica un rendiment equivalent a l'atzar [34].

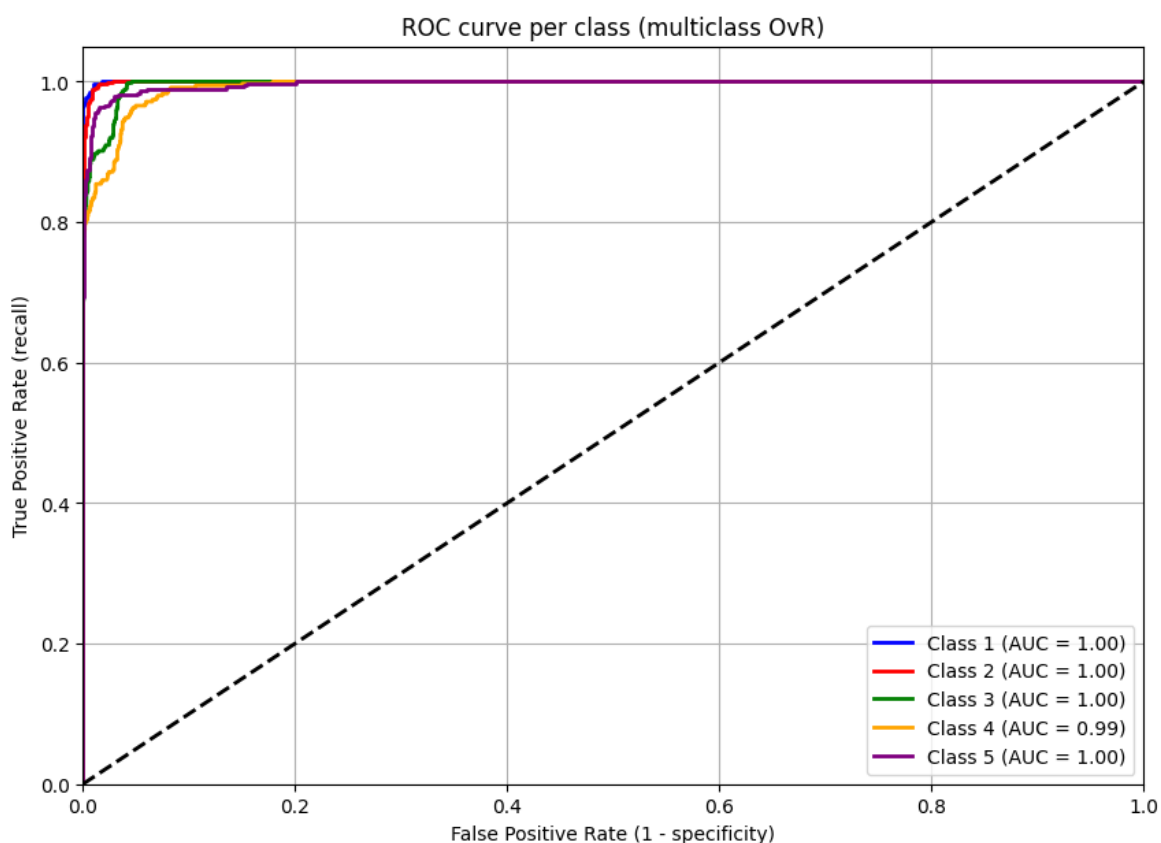


Figura 10. AUC per cadascun dels nivells de fatiga.

El **llindar** és el valor a partir del qual es decideix si una mostra es classifica com a positiva o no, en funció de la probabilitat assignada pel model [34]. En contextos de classificació binària, per exemple, és habitual considerar com a positiu un cas on la probabilitat predita superi el valor de 0.5. No obstant això, en aquest treball s'ha utilitzat una

²⁸ Area Under the Curve

estratègia de **classificació multiclasse basada en l'enfocament OvR**²⁹ [35], on es genera una corba ROC per a cada classe tractant-la com a classe positiva i agrupant totes les altres com a negatives.

En aquest context, **no s'ha aplicat cap llindar explícit**, sinó que la **predicció final** per a cada espectrograma s'ha obtingut seleccionant **la classe amb la probabilitat més alta**. Això implica que, tot i que el model pot tenir un **AUC molt elevat**, això **no garanteix una exactitud perfecta**, ja que l'AUC avalua la qualitat de la **classificació probabilística**, no la precisió final de la predicció [34].

Per exemple:

- Suposem un cas on la classe real és 3, i el model assigna aquestes probabilitats:
[Classe 1: 0.01, Classe 2: 0.02, Classe 3: 0.85, Classe 4: 0.07, Classe 5: 0.05]

El model no només encerta la predicció (classe 3), sinó que assigna una probabilitat molt alta a la classe real, contribuint molt positivament a l'AUC.

- Suposem ara un altre cas on la classe real és 4, però les probabilitats obtingudes són:
[Classe 1: 0.01, Classe 2: 0.01, Classe 3: 0.77, Classe 4: 0.20, Classe 5: 0.01]

En aquest cas, el model fa una predicció incorrecta (classe 3), però assigna una probabilitat relativament alta a la classe real (classe 4). Això **pot seguir sent valorat positivament per l'AUC**, perquè la mètrica considera com de bé el model ordena les probabilitats per a la classe real respecte a les altres, independentment de quina sigui finalment seleccionada com a predicció.

En resum, l'AUC **avalua la capacitat del model per assignar més probabilitat a les etiquetes correctes que a les incorrectes**, però **no reflecteix directament l'encert global del model (accuracy)**. És per això que un model pot obtenir un AUC proper a 1.0 i, tot i així, tenir una taxa d'error no nul·la segons la matriu de confusió.

8.3 Avaluació del Rendiment sobre Mostres Diferents a les Dades Inicials

Amb l'objectiu d'avaluar la capacitat de generalització del model, s'ha predit experimentalment el nivell de fatiga sobre diferents mostres, obtingudes amb el sensor de *Movesense*, a 200 Hz.

Per una banda, s'han analitzat 50 mostres procedents dels ciclistes col·laboradors d'ISGlobal. Aquestes mostres es van obtenir a cegues, és a dir, sense conèixer prèviament el nivell de fatiga ni el moment exacte de la seva captura. Sorprenentment, totes les mostres han mostrat un **nivell de fatiga 1**, equivalent a un estat de repòs, tot i que s'esperava detectar indicis de fatiga postentrenament. Aquesta anomalia es pot explicar per la fisiologia del sistema nerviós autònom, concretament pel paper del **sistema parasimpàtic**.

Després d'un exercici intens, el cos activa el sistema parasimpàtic, conegut com el sistema de "repòs i digestió", que contraresta l'efecte del sistema simpàtic, responsable de la resposta de "lluïta o fugida". Aquesta activació parasimpàtica redueix la freqüència cardíaca

²⁹ *One-vs-Rest*

i promou la recuperació corporal. Estudis han demostrat que la freqüència cardíaca pot disminuir significativament en els primers minuts postexercici, principalment a causa de la reactivació parasimpàtica [36].

En el cas dels ciclistes, les mostres es van prendre quan ja estaven a l'hotel, en un estat de repòs prolongat després de l'entrenament. Aquest període va permetre una recuperació completa del sistema nerviós autònom, especialment del sistema parasimpàtic, que va portar els seus paràmetres fisiològics a nivells basals. Això explica **per què** el model ha detectat un **nivell de fatiga 1** en totes les mostres, reflectint un estat de repòs més que de fatiga activa.

D'altra banda, també s'han analitzat diverses mostres personals, recollides en diferents condicions i moments del dia, amb l'objectiu de posar a prova el rendiment del model en situacions variades. Els resultats obtinguts d'aquestes proves es troben resumits a la *Taula 2*.

Condició	Duració de la mostra (minuts)	Nivell de fatiga predit	% de probabilitat
Estat de repòs, tombat al llit acabat de despertar	5	1	89.23
Estat de repòs, assegut escrivint aquesta memòria	5.30	1	98.14
10 minuts després d'acabar una sessió de moderada intensitat en bicicleta el·líptica	5	1	99.97
10 minuts després d'acabar una sessió d'alta intensitat en bicicleta el·líptica	5	2	59.85
10 minuts després d'acabar un entrenament de <i>crossfit</i> (bastant aeròbic)	5	3	82.59
Justament després d'acabar un entrenament de <i>crossfit</i> (no tant aeròbic)	5	3	81.66
Durant una sessió de <i>jumping jacks</i> i <i>skipping</i>	2	4	96.54
Durant una sessió en bicicleta el·líptica de moderada intensitat (nivell de força 3/10, a ritme constant de 15 km/h)	5	4	99.99
Durant una sessió de <i>running</i>	10	5	61.78
Durant una sessió en bicicleta el·líptica d'alta intensitat (nivell de força 7/10, a ritme constant de 20 km/h)	5	5	95.60

Taula 2. Resultats experimentals de les prediccions en diferents condicions.

9 Conclusions

Aquest treball ha tingut com a objectiu principal el desenvolupament i avaluació d'un model de xarxa neuronal capaç de detectar el nivell de fatiga d'una persona a partir de senyals fisiològics. A través d'un enfocament interdisciplinari, s'ha buscat integrar coneixements de la informàtica, la IA i la salut per oferir una eina amb potencial aplicació pràctica.

Les proves realitzades amb mostres pròpies —tot i tenir un caràcter experimental— han demostrat una notable correspondència amb les condicions reals en què es van enregistrar, reforçant així la capacitat de generalització del model. Això és especialment destacable si es considera que les dades utilitzades provenen de sensors diferents als de l'entrenament original. En general, s'ha observat que el model és capaç d'identificar correctament el nivell de fatiga, amb prediccions coherents amb l'estat físic real.

Ara bé, cal remarcar que, en escenaris concrets com la sessió de *running*, tot i que el nivell de fatiga predit sembla ser el correcte, el percentatge de confiança obtingut és més baix que en altres condicions. Aquest comportament es podria atribuir al soroll afegit durant l'activitat física intensa, el qual pot interferir en la qualitat del senyal i afectar lleugerament la robustesa de la predicció.

Una altra limitació rellevant és la manca de diversitat en la base de dades utilitzada per a l'entrenament. El fet que totes les mostres pertanyin a homes pot generar biaixos significatius, com s'ha pogut constatar amb la predicció feta sobre una noia de 24 anys. Tot i que el model va encertar el nivell de fatiga (nivell 1, repòs), la confiança de la predicció només va ser del 49%. Aquest resultat subratlla la necessitat d'incorporar dades més variades, tant pel que fa al sexe com al tipus de sensors emprats.

Una bona opció per millorar la qualitat de les prediccions futures seria l'ús del *transfer learning*. Aquest mètode consistiria, a partir del model ja entrenat amb les dades existents, adaptar-lo amb noves dades procedents de diferents sexes i sensors. Normalment, en aquests casos, es mantenen intactes les primeres capes convolucionals, que capturen característiques generals com ara patrons bàsics del senyal, i s'ajusten o reentrenen les últimes capes, que són les responsables de la classificació específica. D'aquesta manera, es pot aprofitar al màxim el coneixement adquirit inicialment i ajustar el model per millorar la seva capacitat predictiva amb les noves dades.

Cal remarcar també que, tot i les intencions del projecte, encara no s'ha pogut aplicar el model a dones amb càncer, que era un dels col·lectius d'interès. Aquest fet es deu al fet que, en el moment de dur a terme aquest treball, encara no es disposava del grup definitiu d'aquestes participants ni de les seves dades fisiològiques enregistrades. Per tant, l'efectivitat del sistema en aquest context específic continua pendent de validació.

D'altra banda, pel que fa a l'aplicació mòbil desenvolupada, aquesta representa una primera versió que encara pot evolucionar molt més. L'objectiu futur és executar la CNN en un servidor AWS³⁰, la qual cosa facilitaria el desplegament i l'escalabilitat del sistema per ser usat de manera massiva.

A nivell personal, aquest projecte m'ha permès explorar una mica més la IA i la ciència de dades, àrees que em resulten especialment interessants. A més, he pogut relacionar

³⁰ Amazon Web Service

aquestes disciplines amb la salut, lligant certament amb la carrera de Biotecnologia. Aquesta connexió interdisciplinària m'ha donat una visió més completa i enriquidora de com aplicar la tecnologia per millorar la qualitat de vida i la salut de les persones.

En definitiva, aquest projecte ha aconseguit **resultats molt satisfactoris** i demostra que la IA pot ser una eina molt útil per monitoritzar la fatiga de manera objectiva. Tot i les limitacions actuals, el treball realitzat ha establert una bona base per seguir millorant i avançant en aquesta línia d'investigació.

10 Consideracions Ètiques i de Responsabilitat Social

10.1 Igualtat

El model desenvolupat presenta un **biaix de gènere rellevant**, ja que s'ha entrenat únicament amb dades de participants homes. Això ha tingut conseqüències clares en la precisió i la confiança de les prediccions quan s'ha aplicat a dones, com en el cas de la prova amb una noia de 24 anys, on la classificació era correcta però amb una probabilitat molt baixa (49%). Aquesta situació posa de manifest la importància de tenir en compte la diversitat de gènere en la recollida de dades i entrenament dels models d'IA, per evitar que es perpetuïn discriminacions i desigualtats.

Tot i que aquest biaix ha estat identificat, encara no s'ha pogut corregir plenament per limitacions pràctiques. No obstant això, es proposa com a línia de millora futura l'ampliació de la base de dades per incloure mostres de dones i altres col·lectius infrarepresentats, així com l'ús de tècniques de *transfer learning* per adaptar el model a aquesta diversitat.

10.2 Medi Ambient

L'**impacte ambiental** directe d'aquest projecte és **baix**, ja que principalment es tracta d'un desenvolupament de software i una aplicació experimental a petita escala. No obstant això, s'ha tingut en compte la possibilitat d'**optimitzar l'ús de recursos computacionals**, especialment de cara al desplegament en entorns com AWS. L'eficiència en el consum energètic del model, especialment si s'escalen els serveis a molts usuaris, és un aspecte a considerar per minimitzar l'impacte ambiental. Així mateix, la recollida de dades s'ha fet amb dispositius accessibles i de baix consum, evitant tecnologies invasives o amb una petjada ecològica elevada.

10.3 Responsabilitat Social

Aquest projecte parteix d'una necessitat social clara: la monitorització objectiva de la fatiga per millorar el benestar i la salut, especialment en col·lectius vulnerables com dones amb càncer o ciclistes sotmesos a alts nivells d'esforç. Tot i que no s'ha pogut validar directament en aquests grups, el model pretén ser una **eina d'ajuda**, no invasiva i accessible, amb impacte potencial positiu en la qualitat de vida d'aquestes persones.

10.4 Ètica

S'ha procurat actuar d'acord amb els principis deontològics de la ciència de dades, com ara el respecte per la privadesa, la transparència en el tractament de dades i la no discriminació. Aquest compromís es tradueix en l'ús exclusiu de dades recollides pels investigadors amb **consentiment informat**, on van garantir l'anonimat i van informar els participants sobre la finalitat del projecte i l'ús de les seves dades fisiològiques.

A més, s'ha dut a terme una reflexió crítica sobre com aquestes garanties ètiques poden mantenir-se en una **possible aplicació real del sistema**. Per exemple, s'ha considerat que, per evitar riscos associats a la filtració o mal ús de dades sensibles, caldria implementar **sistemes d'autenticació d'usuari, xifratge de dades i protocols de privadesa reforçats**, a més de mesures legals com la signatura de consentiments digitals.

11 Referències

- [1] M. M. Taye, «Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions», *Computation*, vol. 11, n.º 3, p. 52, mar. 2023, doi: 10.3390/computation11030052.
- [2] B. Roelands, V. Kelly, S. Russell, y J. Habay, «The Physiological Nature of Mental Fatigue: Current Knowledge and Future Avenues for Sport Science», *International Journal of Sports Physiology and Performance*, vol. 17, n.º 2, pp. 149-150, feb. 2022, doi: 10.1123/ijsp.2021-0524.
- [3] B. Pedraz-Petrozzi, «Fatiga: historia, neuroanatomía y características psicopatológicas. Una revisión de la Literatura.», *Rev Neuropsiquiatr*, vol. 81, n.º 3, p. 174, oct. 2018, doi: 10.20453/rnp.v81i3.3385.
- [4] I. Völker, C. Kirchner, y O. L. Bock, «On the relationship between subjective and objective measures of fatigue», *Ergonomics*, vol. 59, n.º 9, pp. 1259-1263, sep. 2016, doi: 10.1080/00140139.2015.1110622.
- [5] N. R. Adão Martins, S. Annaheim, C. M. Spengler, y R. M. Rossi, «Fatigue Monitoring Through Wearables: A State-of-the-Art Review», *Front. Physiol.*, vol. 12, p. 790292, dic. 2021, doi: 10.3389/fphys.2021.790292.
- [6] H. Al-Libawy, A. Al-Ataby, W. Al-Nuaimy, y M. A. Al-Tae, «HRV-based operator fatigue analysis and classification using wearable sensors», en *2016 13th International Multi-Conference on Systems, Signals & Devices (SSD)*, Leipzig, Germany: IEEE, mar. 2016, pp. 268-273. doi: 10.1109/SSD.2016.7473750.
- [7] L. M. D. Araújo, C. M. S. D. Carvalho, M. M. Amaral, y L. D. Santos, «Variabilidade da Frequência Cardíaca como biomarcador do estresse: revisão integrativa», *RSD*, vol. 9, n.º 12, p. e34991211125, dic. 2020, doi: 10.33448/rsd-v9i12.11125.
- [8] A. Di Costanzo, C. A. M. Spaccarotella, G. Esposito, y C. Indolfi, «An Artificial Intelligence Analysis of Electrocardiograms for the Clinical Diagnosis of Cardiovascular Diseases: A Narrative Review», *JCM*, vol. 13, n.º 4, p. 1033, feb. 2024, doi: 10.3390/jcm13041033.
- [9] «Movesense». [En línea]. Disponible en: <https://www.movesense.com/>
- [10] A. Kazemnejad, P. Gordany, y R. Sameni, «EPHNOGRAM: A Simultaneous Electrocardiogram and Phonocardiogram Database». *PhysioNet*. doi: 10.13026/TJQTQ-5911.
- [11] S. M. Debbal y F. Bereksi-Reguig, «Computerized heart sounds analysis», *Computers in Biology and Medicine*, vol. 38, n.º 2, pp. 263-280, feb. 2008, doi: 10.1016/j.combiomed.2007.09.006.
- [12] Y. Jia *et al.*, «Preprocessing and Denoising Techniques for Electrocardiography and Magnetocardiography: A Review», *Bioengineering*, vol. 11, n.º 11, p. 1109, nov. 2024, doi: 10.3390/bioengineering11111109.
- [13] B. Lenka, «Time-frequency analysis of non-stationary electrocardiogram signals using Hilbert-Huang Transform», en *2015 International Conference on Communications and Signal Processing (ICCSP)*, Melmaruvathur, India: IEEE, abr. 2015, pp. 1156-1159. doi: 10.1109/ICCSP.2015.7322686.
- [14] S. Elouaham, R. Latif, A. Dliou, M. Laaboubi, y F. M., «Parametric and Non Parametric Time-Frequency Analysis of Biomedical Signals», *IJACSA*, vol. 4, n.º 1, 2013, doi: 10.14569/IJACSA.2013.040110.
- [15] «VGG: ¿Qué es este modelo?» [En línea]. Disponible en: <https://datascientest.com/es/vgg-que-es-este-modelo-daniel-te-lo-cuenta-todo>
- [16] C. Cai *et al.*, «Transfer Learning for Drug Discovery», *J. Med. Chem.*, vol. 63, n.º 16, pp. 8683-8694, ago. 2020, doi: 10.1021/acs.jmedchem.9b02147.
- [17] «Early Stopping». [En línea]. Disponible en: https://keras.io/api/callbacks/early_stopping/
- [18] «Learning Rate Scheduler». [En línea]. Disponible en: https://keras.io/api/callbacks/learning_rate_scheduler/
- [19] M. Mateo-March, M. Moya-Ramón, A. Javaloyes, C. Sánchez-Muñoz, y V. J. Clemente-Suárez, «Validity of detrended fluctuation analysis of heart rate variability to determine intensity thresholds in elite cyclists», *European Journal of Sport Science*, vol. 23, n.º 4, pp. 580-587, abr. 2023, doi: 10.1080/17461391.2022.2047228.
- [20] «Python 3.11.4». [En línea]. Disponible en: <https://www.python.org/downloads/release/python-3114/>
- [21] «NumPy 2.1.3». [En línea]. Disponible en: <https://numpy.org/devdocs/release/2.1.3-notes.html>
- [22] «Matplotlib 3.9.2 documentation». [En línea]. Disponible en: https://matplotlib.org/3.9.2/api/_as_gen/matplotlib.axes.Axes.step.html
- [23] «seaborn 0.13.2 documentation». [En línea]. Disponible en: <https://seaborn.pydata.org/installing.html>
- [24] «pandas 2.2.3 documentation». [En línea]. Disponible en: <https://pandas.pydata.org/docs/>
- [25] «neurokit2 0.2.11». [En línea]. Disponible en: <https://pypi.org/project/neurokit2/>
- [26] «keras 3.8.0». [En línea]. Disponible en: https://keras.io/keras_3/
- [27] «scikit-learn 1.5.2». [En línea]. Disponible en: <https://pypi.org/project/scikit-learn/1.5.2/>
- [28] «ecg_clean()». [En línea]. Disponible en: <https://neuropsychology.github.io/NeuroKit/functions/ecg.html>

- [29] M. Kumngern, N. Aupithak, F. Khateb, y T. Kulej, «0.5 V Fifth-Order Butterworth Low-Pass Filter Using Multiple-Input OTA for ECG Applications», *Sensors*, vol. 20, n.º 24, p. 7343, dic. 2020, doi: 10.3390/s20247343.
- [30] R. E. Challis y R. I. Kitney, «The design of digital filters for biomedical signal processing Part 3: The design of Butterworth and Chebychev filters», *Journal of Biomedical Engineering*, vol. 5, n.º 2, pp. 91-102, abr. 1983, doi: 10.1016/0141-5425(83)90026-2.
- [31] «Flutter». [En línea]. Disponible en: <https://docs.flutter.dev/install/archive>
- [32] «mdsflutter». [En línea]. Disponible en: <https://www.movesense.com/news/2020/08/another-flutter-plugin-for-movesense-to-support-both-ios-and-android/>
- [33] «Flask». [En línea]. Disponible en: <https://pypi.org/project/Flask/>
- [34] Ş. K. Çorbacıoğlu y G. Aksel, «Receiver operating characteristic curve analysis in diagnostic accuracy studies: A guide to interpreting the area under the curve value», *Turkish Journal of Emergency Medicine*, vol. 23, n.º 4, pp. 195-198, oct. 2023, doi: 10.4103/tjem.tjem_182_23.
- [35] «OneVsRestClassifier». [En línea]. Disponible en: <https://scikit-learn.org/stable/modules/generated/sklearn.multiclass.OneVsRestClassifier.html>
- [36] G. L. Pierpont, S. Adabag, y D. Yannopoulos, «Pathophysiology of Exercise Heart Rate Recovery: A Comprehensive Analysis», *Noninvasive Electrocardiol*, vol. 18, n.º 2, pp. 107-117, mar. 2013, doi: 10.1111/anec.12061.