Estudi del rendiment mental a Barcelona

Grau d'Intel·ligència Artificial: Xarxes Neuronals i Deep Learning

Adrià Cantarero Carreras i Pau Hidalgo Pujol

7 de maig de 2024

Índex

1 Introducció

El propòsit de la pràctica era poder aplicar el camp de l'aprenentatge automàtic a un cas més realista. En el treball descrit a continuació, es proposa una implementació de models lineals i perceptrons multicapa per tal de modelar el rendiment d'un test d'atenció (per tant, rendiment mental) en funció de diverses variables com la contaminació. El conjunt de dades prové de l'estudi CitieS-Health Barcelona realitzat entre finals del 2020 i principis del 2021 [GRT+22a], i conté informació sobre l'estat físic i mental de diverses persones en un període de temps, així com dades de contaminació de l'entorn on viuen. A partir d'aquestes dades, que s'exploraran a continuació, es buscarà intentar determinar si es poden utilitzar aquests factors per predir el rendiment mental d'una persona.

2 Anàlisi exploratori de les dades

Per començar, en aquest primer apartat tractem les diferents anàlisis fetes a les variables, per a entendre amb què estem treballant, i com ho tractarem posteriorment.

2.1 Anàlisi univariant

Primerament, descriurem la nostra base de dades breument.

La *database* presenta 3348 files i 95 variables (columnes). Algunes d'aquestes variables les transformem a categòriques ja en aquest punt (les variables transformades es poden veure al *notebook*). A més, també tenim moltes variables redundants o que depenien d'altres (alta correlació), que hem directament eliminat. Després d'aquest petit tractament, de les 95 variables, ens quedem només amb 55. D'aquestes, 21 són numèriques i 34 categòriques.

Després d'això, al *notebook* mostrem les distribucions de les variables numèriques i els gràfics de barres de les categòriques que, per qüestió d'espai no podem mostrar aquí, però que es troben visibles al *notebook*. Per les numèriques, veiem des de distribucions normals a distribucions totalment exponencials (les quals escalarem posteriorment); mentre que per les categòriques tenim des de variables binàries (dues categories), fins a variables de múltiples categories, superior a deu en alguns casos.

2.2 Anàlisi de correlacions

Per aquesta anàlisi, hem creat una matriu de correlació, per a poder observar les correlacions entre variables numèriques de la *database* (vegeu figura 1).

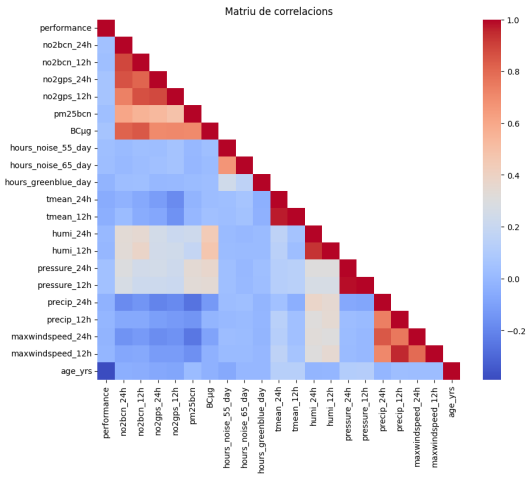

Figura 1: Matriu de correlacions entre variables numèriques

En aquest gràfic, podem veure com la majoria de les variables no tenen una correlació positiva significativa. Això no obstant, hi ha dos petits grups que presenten altres correlacions positives entre les variables d'aquest. El primer fa referència a tots els contaminants mesurats, ja que les variables de *NO2* varien només en les hores de mesura. Passa el mateix amb l'altre grup, referent a les variables de precipitació i velocitat del vent. Encara que s'entén la correlació entre les dues, les altres correlacions també són entre la mateixa magnitud en diferents rangs horaris de mesures). Per això, farem un procés de selecció d'aquestes variables a la secció 3.3.

Per altra banda, tenim una correlació significativament negativa, la qual relaciona la variable d'edat (*age_yrs*) amb el rendiment mental de la persona (*performance)*, la nostra variable a predir (vegeu secció 4). Si analitzem aquesta relació en un *lineplot* (vegeu *notebook*), podem veure com contra més edat, la *performance* tendeix a decréixer. Encara que observem una certa oscil·lació durant totes les edats, a partir dels 65 anys la *performance* cau en picat, sense tornar a pujar.

2.3 ANOVA

Per aprofundir encara més en les relacions entre les variables, hem decidit realitzar una ANOVA de la variable *performance* respecte a totes les variables categòriques existents.

A la taula 1 podem veure totes aquelles variables categòriques que presenten un *p-*

Variable	sum_sq	df	F	PR(¿F)
estres	6815.227	10.0	6.484368	5.738416e-10
ordenador	2655.953	1.0	25.036952	5.956899e-07
otrofactor	913.822	1.0	8.588159	0.003410
occurrence_stroop	32515.594	13.0	25.582554	1.004899e-59
precip_24h_binary	539.053	1.0	4.996360	0.025473
incidence_cat	1829.716	3.0	5.744201	0.000648
smoke	494.207	1.0	4.620204	0.031686
district	11766.528	10.0	11.440346	1.794005e-19
education	866.628	2.0	4.020372	0.018052
covid_work	4900.029	4.0	11.455512	3.142146e-09
covid_mood	5376.934	4.0	12.581629	3.773107e-10
covid_sleep	1393.530	4.0	3.213705	0.012134
covid_espacios	1432.655	2.0	6.628483	0.001343
covid_aire	2098.456	2.0	9.729952	0.000061
covid_motor	2039.438	2.0	9.450547	0.000081
covid_electric	1216.353	2.0	5.686426	0.003432
covid_bikewalk	2303.300	2.0	10.669158	0.000024
covid_public_trans	760.856	2.0	3.506744	0.030124

Taula 1: Taula de resultats amb *p-value* < 0.05 del test ANOVA

value menor a 0.05. És a dir, que la variable *performance* presenta diferències estadísticament significatives entre les categories de la variable categòrica.

Observem algunes variables de caràcter personal, com ara *estres*, *ordenador*, *smoke* o *education* (al *notebook* es presenten gràfics de *boxplots* comparant les categories). També veiem diferències significatives de rendiment entre districtes (vegeu 2.6). Finalment, sembla que totes les variables referents a les afectacions a la vida personal per l'etapa de *COVID* tenen diferències de rendiment mental depenent de les categories d'aquestes.

2.4 Detecció d'outliers

Arribats a aquest punt, fem una anàlisi d'outliers utilitzant el mètode *Isolation Forest*. Podem observar com, en aquestes dimensions, tenim outliers visibles com a grups fora de la "mitjana" observable.

Si visualitzem els components del PCA en un altre gràfic (vegeu *notebook*), podem veure com els que tenen més força són els referents a les variables de precipitació i humitat; així com les variables de contaminació (de *NO2*). Això és probablement relacionat de la següent manera: els dies de precipitació, la humitat augmenta, i tot això provoca un decreixement de la contaminació, creant aquests suposats outliers que veiem [...]. A causa de això, hem decidit no eliminar aquests outliers, ja que són creats de manera natural. És a dir, encara que és cert que són valors extrems que s'allunyen significativament de la mitjana, no han sigut mesurats així degut a errors o un mal funcionament del sensor.

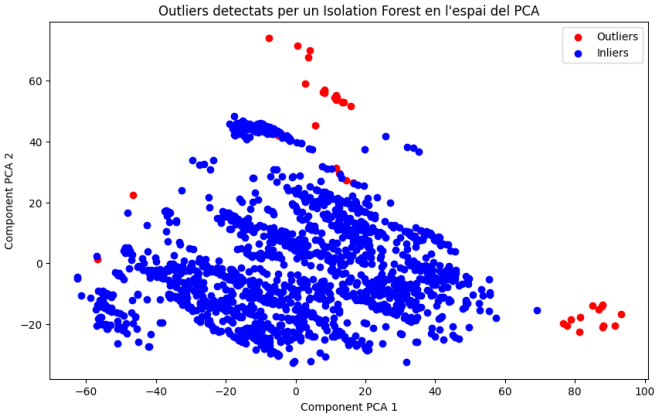

Figura 2: Outliers detectats per un *Isolation Forest* en l'espai del PCA

2.5 Anàlisi de missing values

En aquest apartat, fem una anàlisi dels missing values, i discutim com els tractarem a la secció de preprocessament (vegeu apartat 3.2). Al *notebook* es pot veure una taula amb tots els missing values per variable. En aquesta, podem observar com en la majoria de les variables tenim quantitats entre 100 i 300 missings. Això no obstant, aquests missings poden coincidir a les mateixes files, tenint files amb una quantitat de missings exagerada. És per això, que fem el barplot 3, on representem la quantitat de missing values per fila. Això ho fem per analitzar si tenim files amb més d'un cert *threshold* de missing values i eliminar-les (vegeu apartat 3.2).

Al gràfic 3 podem veure com la majoria de les files presenten quantitats de missing values relativament baixes. Tanmateix, tenim casos de files que presenten fins i tot més de 10 missing values per fila (recordar que tenim 55 columnes més la variable a predir *performance*).

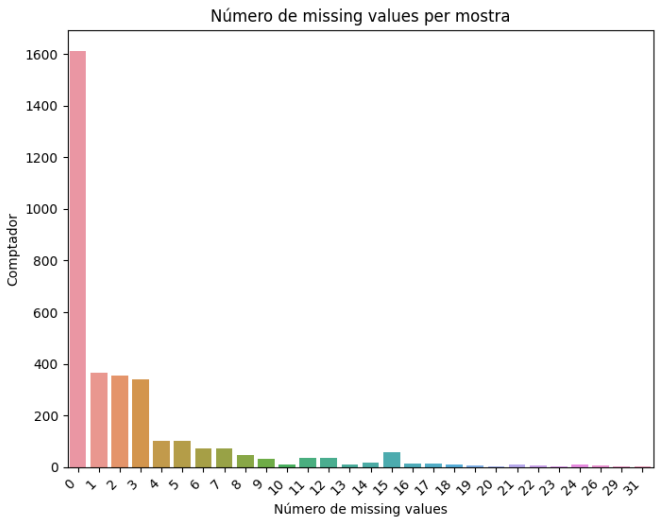

Figura 3: Compteig de missing values per fila

2.6 Anàlisi multivariable geoespacial

Com hem vist al test ANOVA (vegeu apartat 2.3), podem confirmar que la variable a predir *performance* presenta diferències estadísticament significatives a la variable de districtes. És per això que hem decidit representar els rangs de valors per aquesta variable en un mapa dels districtes de Barcelona. A més, també hem cregut interessant representar altres variables de caràcter personal així com de contaminació. D'aquesta manera, podem fer una anàlisi multivariable per districtes de Barcelona, observant les diferències que presenten.

A la figura 4, podem veure les variables *age_yrs*, *BCμg*, *bienestar*, *estres*, *no2BCN_12h*, *performance*, *precip_12h* i *sueno* representades per rangs de valors en els

diferents districtes de Barcelona. És d'interès remarcar que les nostres dades fan referència als pacients de l'estudi, de manera que aquestes representacions no busquen ni molt menys extreure conclusions de les característiques de la població total de Barcelona.

Per l'edat (*age_yrs*), podem veure com els pacients més joves es troben al districte de Nou Barris; mentre que els pacients d'edats més avançades resideixen a Les Corts. Per la contaminació de *black carbon* ($BC\mu g$), observem com els districtes amb més concentració d'aquest contaminant són Les Corts, Sant Andreu i Sants-Montjuïc. Pel benestar dels pacients (*bienestar*), veiem valors més alts a districtes com Les Corts i Sarrià-Sant Gervasi; mentre que districtes com Horta-Guinardó i Nou Barris presenten pacients amb menys benestar. Per l'estrès dels pacients (*estres*), els districtes amb pacients més estressats són Nou Barris i Sants-Montjuïc, mentre que Les Corts presenta els valors més baixos, entre d'altres. Per la contaminació de NO2 (*no2BCN_12h*), tenim les mesures més altes a Sant Andreu i Sants-Montjuïc, i les més baixes a Sarrià-Sant Gervasi i Ciutat Vella. Pel rendiment mental (*performance*), el valor més alt el presenta Sant Andreu, mentre que Nou Barris presenta els pitjors valors. Per la precipitació (*precip_12h*), tenim valors bastant semblants (i alts) a tots els districtes, menys a Gràcia i Ciutat Vella, on els valors són menors. Per acabar, els valors de la qualitat de son (*sueno*) més alts els té Les Corts i Ciutat Vella; mentre que els pacients que descansen pitjor resideixen a Horta-Guinardó i Nou Barris.

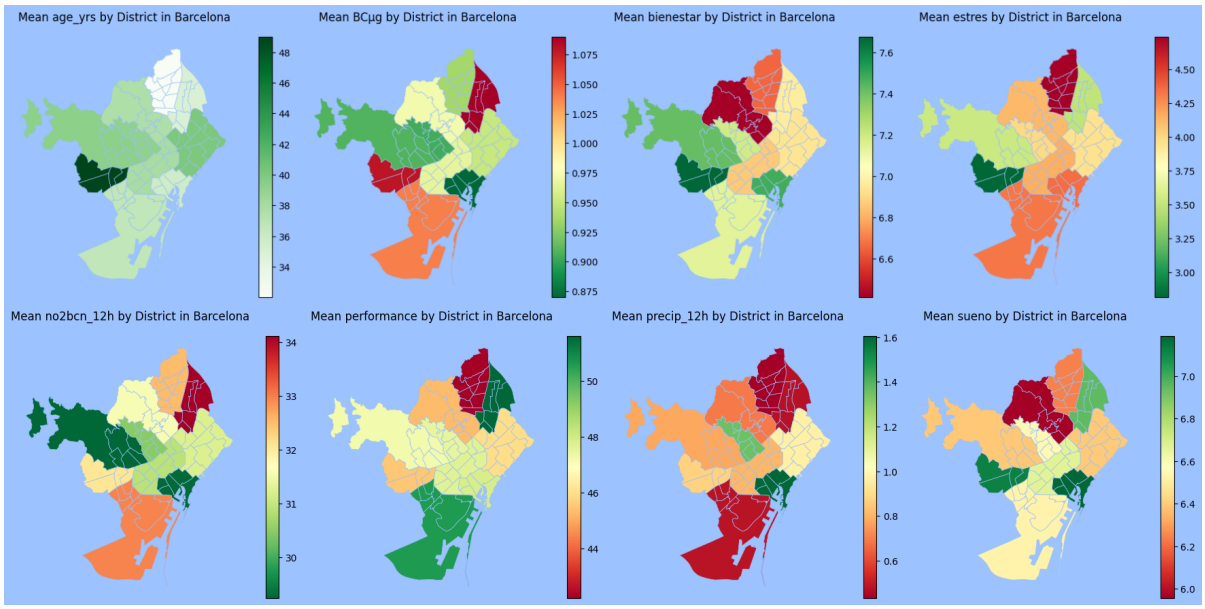

Figura 4: Mapa dels districtes de Barcelona per diverses variables

3 Preprocessament

En aquesta secció, definim breument els diferents passos de preprocessament realitzats.

3.1 Tractament previ de missing values

Abans de realitzar la partició de train i test, hem tractat els missing values per a tota la *database*. Primerament, hem esborrat tots els missing values per a la variable *performance*, ja que és la variable que volem predir. Després, hem esborrat totes les files que presentaven més de 10 missings (per aquella fila) que hem vist al gràfic 3, ja que seria poc fiable imputar aquests valors segons tan pocs valors existents en aquella fila. Arribats a aquest punt, teníem encara variables que presentaven valors des de 100 a 400 missing values (en tota la *database* per aquella variable). Hem decidit esborrar directament totes aquelles variables amb més de 280 missing values, puix que seria poc fiable imputar valors per una variable la qual té més d'un 10% de valors inexistents.

3.2 Imputació de missing values

A partir d'aquesta secció, **ja treballem amb la partició en train i test set**, realitzada amb una proporció 80/20%, respectivament. Després d'haver tractat els missing values a la secció anterior (vegeu secció 3.1), en aquest

apartat prosseguim amb la imputació dels missing values restants, utilitzant tècniques diferents per imputar les variables numèriques i categòriques.

Per a les variables numèriques, hem usat l'imputador iteratiu *IterativeImputer*, de la llibreria *sklearn* [PVG+11], de *Python*, que és multivariant. Per altra banda, per la imputació de variables categòriques hem implementat un algorisme de KNN (*k-nearest neighbors*) que imputa per la moda de les variables.

3.3 Feature selection i transformation

En aquest apartat, simplement tractem les variables que presentaven altes correlacions (vegeu apartat 2.2), fusionant-les o bé eliminant-les.

Pel que fa a les variables meteorològiques i de contaminació (vistes a la matriu de correlacions), simplement ens hem quedat amb les variables que tenien en compte la mesura de dotze hores anteriors ("variable_12h"), ja que hem considerat que era una mesura més recent. Per les variables del contaminant NO2 també hem escollit la referent a la mesura per dades de GPS, car hem considerat que era més precisa que la de les estacions.

Per altra banda, anteriorment hem vist la gran quantitat de variables textuals sobre COVID presents a la *database*. Per aquestes hem decidit fusionar-les i reduir-les a una sola variable binària *covid_afecta*. Per a arribar a aquesta variable, primer hem transformat totes les variables textuals referents a la COVID com binàries. Per fer tal cosa, hem assumit com a *Yes* si en aquella categoria al pacient sí l'ha afectat la COVID (hi ha coses que han canviat a la seva vida), i com a *No* si no ha afectat (o s'ha quedat igual que abans per algunes categories). Per acabar, realitzem la moda d'aquestes variables binàries per a cada individu. Si la moda és *Yes*, aquest serà el valor per a la nova variable *covid_afecta*; i si la moda és *No*, doncs el contrari.

3.4 Recodificació de variables categòriques

Per tal que els nostres models puguin treballar amb dades categòriques, calia convertir-les d'alguna manera. Hem optat per realitzar OneHot encoding en elles, excepte en les variables que tenen categories ordinals (com per exemple estrès). En aquests casos, hem mantingut els nombres originals, ja que representen bé l'ordre i té sentit usar-los.

3.5 Estandarització

Finalment, per tal que el model pugui treballar de manera més senzilla amb les dades, hem fet una estandarització de les dades utilitzant Standard Scaler. Per evitar data leakage i mantenir la coherència de les particions, s'entrena amb l'entrenament, i el test simplement es transforma.

4 Models

Un cop preprocessades les dades i seleccionades les més adients, calia procedir amb la creació dels models. La variable escollida com a objectiu és *performance*, que permet estimar el rendiment mental en funció dels resultats del test de Stroop. En tenir aquests valors numèrics, el problema és un de regressió. La mètrica emprada a partir d'aquest punt per avaluar els models serà l'R2.

4.1 Regressió lineal

D'entrada, per tal de dur a terme la regressió lineal correctament, cal tenir en compte les assumpcions que fa: linearitat, independència de les observacions, normalitat dels residus, no multicol·linealitat (correlacions massa elevades entre variables) i homoscedasticitat. Les que podem comprovar de forma més senzilla són les de linearitat i normalitat dels residus, i hem vist que es compleixen les dues.

D'entrada, hem provat la regressió lineal tal com està implementada a la llibreria *sklearn* [PVG+11]. La implementació és realment senzilla: entrenem amb el train, realitzem una cross validation (amb 10 talls) per obtenir un valor de validació i finalment, realitzem les prediccions amb el test. Els valors de r2 obtinguts són:

R2 train:	0.3227237347301225
R2 val:	0.28776214019996044
R2 test:	0.28539387476445643

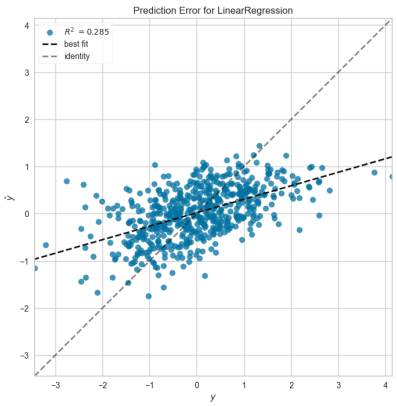

Figura 5: Regressió lineal

Com es pot observar, l'R2 que hem assolit en el test és de 0.2837. Tot i això, aquest queda lleugerament per sota el del train. El Mean Squared Error és 0.68 en el train, comparat amb 0.78 en el test. Els resultats de la regressió es poden observar a la figura 5

La versió per defecte de la regressió linear no té paràmetres. Tot i això, es poden incloure regularitzacions amb alguns models diferents: el ridge regression (l2), el Lasso Regression (l1) i ElasticNet (combinació dels dos anteriors). Aquests models estan també implementats a sklearn [PVG+11]. Cal mencionar que tenen paràmetres que s'han d'ajustar (les alphas o lambdas), i que, per tant, s'ha fet servir una versió de la implementació que utilitza cross-validation per aconseguir el millor d'aquests valors.

Podem observar els resultats en el test, comparats també amb els del linear regression, a la següent taula:

	LR	Ridge	Lasso	Elasticnet
lambda	0.000000	10.000000	0.007205	0.012532
Train R2	**0.322724**	0.322707	0.318300	0.318780
CV (Val) R2	0.287762	**0.287988**	0.287575	0.287667
Test R2	0.285394	**0.285396**	0.282252	0.282599

Veiem com tots obtenen resultats molt similars, tot i que el ridge regression (el que implementa regularització l2) obté un validation lleugerament més bo (el test està afegit, però no ens podem basar en ell per determinar quin model és millor). En general, però, sembla que no es poden millorar gaire aquests resultats usant un model de regressió lineal (pot variar molt lleugerament en funció del solver utilitzat o la implementació usada, però valors insignificants)

A resultats, analitzarem els coeficients d'aquests models de regressió, ja que ens poden permetre interpretar aquests resultats millor i contextualitzar-los.

4.2 MLP (Multilayer Perceptron)

Com hem vist, la regressió lineal sembla que dona valors entorn del 0.28 de variància explicada (R2). Tot i això, cal entendre que un model lineal està limitat a una funció lineal de les dades (o la que permeti la funció d'enllaç), i que, per tant, hi ha alternatives millors, com per exemple els perceptrons multicapa.

Aquests models poden tenir moltes configuracions diferents. El procés per seleccionar la millor l'hem dividit en diverses iteracions, on a cada una s'analitzen els resultats del model per poder millorar.

Dir que, si bé es comentarà tan sols un resultat, s'han realitzat 5 execucions de cada model amb la configuració esmentada per tal de corroborar que els resultats no fossin deguts a factors aleatoris. A més, la mateixa llibreria de tensorflow, en entrenar les capes, ja utilitza una validació interna i per aquest motiu no s'ha creat una tercera partició.

4.2.1 Primera iteració

D'entrada, hem partit d'un model molt senzill, d'una sola capa (més la de sortida) de 50 neurones. En total, té 2651 paràmetres entrenables. El learning rate d'entrada és 0.001, amb l'optimizer Adam, el loss de MSE i un simple callback de Early Stopping.

Tarda tan sols uns 15 segons a entrenar, i obté un 0.7342 de R2 al conjunt d'entrenament, molt més elevat que el de la regressió lineal. Tot i això, cal contextualitzar aquest error: si observem els valors de la validació que ens proporciona tensorflow (keras, [AAB+15]), veiem com aquesta partició té un R2 de tan sols 0.4136 (en un cas únic 0.441234), i un MSE del doble: 0.5262 versus un 0.2624 d'entrenament.

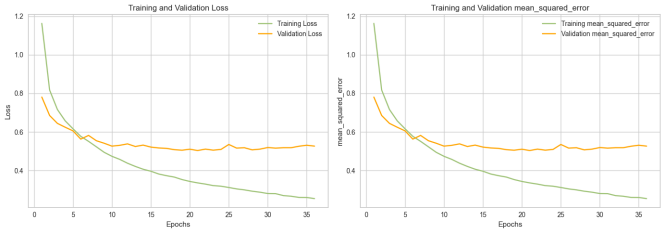

Figura 6: Evolució del loss i l'MSE en l'entrenament

Amb això, podem de moment afirmar que els perceptrons multicapa són ja millors que la regressió lineal, però que de moment sembla que realitza cert overfitting, com es pot en les gràfiques d'evolució de l'entrenament 6. Es veu com el loss de validació, fins i tot augmenta a mesura que hi ha més iteracions.

4.2.2 Segona iteració

Amb vista al succeït en la primera iteració, i per intentar aconseguir que el model no realitzi un overfit tan gran i millorar els resultats de validació, vam implementar dues tècniques de regularització: l2 regularization (com fèiem en la regressió, Ridge) i un lleuger dropout ("apagar" algunes neurones per també permetre una millor generalització. El learning rate, el batch size i la resta de paràmetres semblen estar correctes.

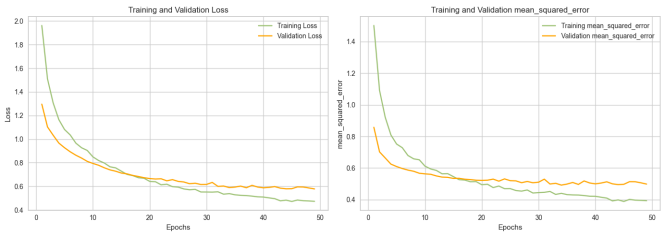

Figura 7: Evolució de l'entrenament de la segona iteració

Amb aquests canvis, hem assolit un R2 d'entrenament més petit, entorn del 0.68, així com un MSE més elevat de 0.33, però en fixar-nos en la validació, hem observat com ha millorat, passant de 0.4412 a 0.4705, i un MSE de menys de 0.5.

Les gràfiques de la validació i l'entrenament (7), en aquest cas, estan molt més juntes. Tot i això, es pot veure com la validació s'estanca bastant (així com l'entrenament), tot i que almenys no augmenta. En general, sembla que la regularització ha ajudat molt, i ja és un bon model.

4.2.3 Tercera iteració

Amb vista als resultats de la segona iteració, hem optat per implementar un model una mica més complex (tenint en compte també que s'haurà de regularitzar), per intentar captar més patrons. Hem escollit un model amb 3 capes: una de 200, una de 100 i una de 50, les tres amb funció d'activació ReLu. La primera no té regularització, mentre que les altres dues sí. A més, hi ha *dropouts* entre totes les capes.

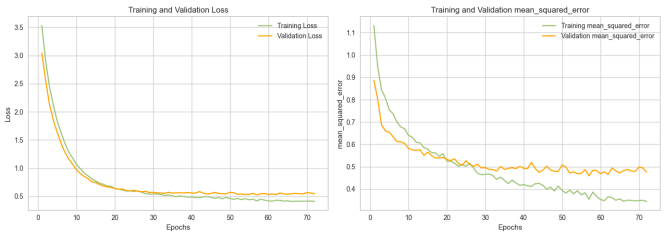

Figura 8: Evolució de la tercera iteració de l'MLP

Els resultats de l'entrenament amb aquest model són un MSE de 0.2387 i R2 de 0.7613 en l'entrenament, mentre que un MSE de 0.47 i un R2 de 0.4954 en la partició de validació (en les 5 repeticions els valors són similars). Observem, doncs, que tot i les regularitzacions els resultats d'entrenament són millors, fins i tot més que la primera iteració. Tot i això, els resultats en la validació són positius. S'analitza a la figura 8 com s'estanca, similar a la iteració anterior.

4.2.4 Quarta iteració

Després de l'anterior iteració, hem dut a terme diverses proves addicionals, que per temes d'espai i per no allargar massa l'explicació no es poden incloure en el document. Aquestes proves han consistit a provar funcions d'error diferents (comentarem l'escollida), així com altres funcions d'activació, mides de batch diferents (més o menys determinista) o regularitzacions d'altres tipus. Aquesta quarta iteració, doncs, busca ser la millor versió del model.

El model final, en lloc d'usar el valor quadràtic mitjà com a funció d'error (loss), utilitza l'error de huber [Hub64]. Aquesta funció penalitza menys els errors molt grans (els outliers), i permet fer una regressió més robusta. També inicialitza el kernel fent servir glorot uniform (també anomeanda xavier, [GB10]). Com que en altres iteracions haviem vist que duia a terme cert overfitting, també incorpora regularització l2 als kernels, i dropout entre les capes (així com normalització, per estandaritzar els inputs). Fa servir 4 capes, amb poques neurones (no més de 80). Un fet curiós de la xarxa és que la penúltima capa conté 100 neurones (la d'abans de la neurona de sortida), amb una ReLu limitada a 1. D'aquesta manera, el valor màxim de sortida serà 100. També implementa un scheduler de learning rate, que comença fent un warmup, i després intenta evitar estancar-se. Igualment, hem canviat el batch size a 64, tot i que mantenint el learning rate. Probablement, es podria portar a cap una xarxa amb menys capes o regularitzacions que realitzés el mateix, però no l'hem aconseguit trobar (tot i que sí algunes que s'acostaven molt a aquests resultats).

Aquest model, en alguns casos, arribava a tenir fins i tot més de 0.50 de R2, tant al validation com al test (al ser el model final, podem realitzar-lo, ja que no el farem servir per seleccionar el millor model).

Veiem a la figura 9 com, tot i que evidentment el train obté millors resultats que el validation, aquest segon continua disminuint, fet positiu i que indica que realment no estem fent overfitting i generalitza al validation. Aquest model s'ha entrenat durant més epochs, i no decreix tan ràpidament el seu loss.

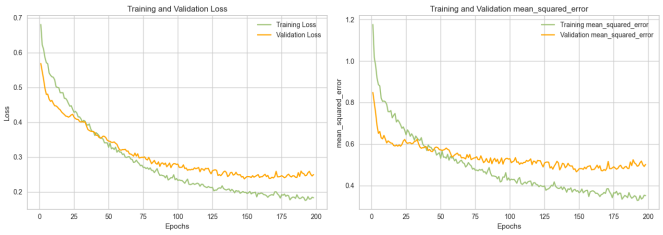

Figura 9: Evolució de la quarta iteració

5 Resultats

De la regressió lineal, podem observar i analitzar-ne els coeficients:

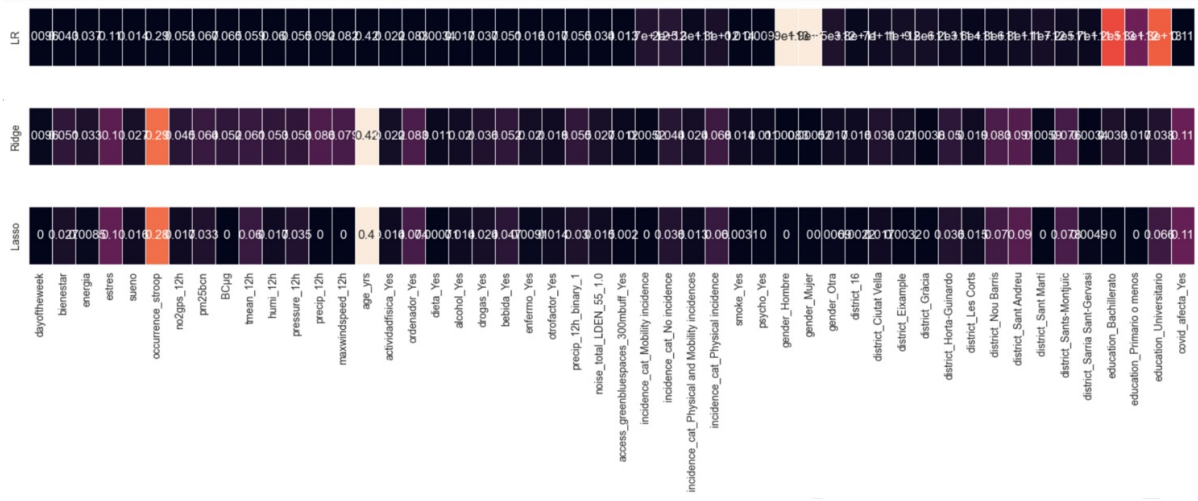

Figura 10: Coeficients de les regressions lineals

Es pot veure una diferència significativa entre els coeficients del model sense regularització i els altres dos. El primer dona molta importància al gènere i a l'educació de les persones, mentre que la resta de variables obtenen coeficients molt propers a zero. Aquests coeficients no són gaire interessants per extreure conclusions, ja que no ens relacionen la performance amb les variables que més ens interessaven (contaminació, estat físic...) sinó amb característiques de les persones que han realitzat el test. Si analitzem els resultats de la mateixa regressió lineal, però en aquest cas de la llibreria statsmodels, veiem com les variables a les quals dona importància són com les del Ridge i del Lasso (i arriba a un r2 de test de 0.2838, lleugerament per sobre).

Aquests altres coeficients són molt més interessants. Veiem com el més important és l'edat, fet que té bastant sentit, ja que a mesura que ens fem grans perdem capacitat cognitiva, i a més es relaciona amb les correlacions vistes en l'EDA. També observem com té una gran influència *occurrence_stroop*, és a dir, quants cops han fet ja el test, fet que pot indicar que les persones es tornen millors resolent aquesta prova amb la pràctica. L'estrès percebut és també una variable important en aquest model, i de fet si analitzem els coeficients veiem com té un efecte negatiu. També són significants el fet de passar-se més de 5 hores davant d'un ordinador, que t'hagi afectat la covid a la teva vida o el barri en el qual viuen. La importància d'aquesta última variable ens podria indicar que hi ha altres característiques en funció del barri que també podrien afectar, però que no han estat recollides en l'estudi.

Pel que fa a les variables meteorològiques, en el cas del Ridge considera lleugerament la concentració de pm25 a Barcelona, així com que hi hagi hagut precipitació en eles últimes 12 hores i forts vents. En general, però, aquestes variables tenen poca importància, tot i que més que altres com fumar, el gènere, prendre alcohol o drogues o la disponibilitat de zones verdes a prop de casa. Com sabem, el Lasso porta a 0 aquelles variables no importants, que en aquest cas són el dia de la setmana, el carboni negre, la pluja (tot i que el Ridge sí que la tenia bastant en compte), el gènere, o l'educació i alguns districtes.

Pel que fa al MLP, és més complicat analitzar els seus resultats i determinar exactament a quins patrons es deuen. Els resultats són molt millors als de la regressió lineal, o sigui que troba certs patrons no lineals en les dades. Tot i això, és complicat poder ajustar el problema de forma que a més, generalitzi, ja que en gairebé tots els casos que hem provat la partició d'entrenament obtenia molts millors resultats que la de validació (o la de test). A més, si regularitzàvem massa, el rendiment del model disminuïa, indicant que és bastant probable que no es puguin aconseguir resultats millors. Tot i això, hem pogut veure com anaven millorant aquests valors amb cada canvi i iteració. Podem observar-ho a la següent taula:

Iteració	**Train R2**	**Validation R2**
Primera iteració	0.734	0.414
Segona iteració	0.687	0.471
Tercera iteració	0.711	0.492
Quarta iteració	0.739	0.510

Taula 2: Mitjana dels R2 màxims de cada iteració.

6 Conclusions

En general, hem pogut observar com realment hi ha certa relació entre el rendiment mental i les dades de les quals disposem. Les de contaminants, si bé no són les més importants, tenen certa rellevància. De fet, en l'estudi del qual provenen les dades [GRT+22b], determinaven de forma estadística aquests efectes adversos, i hi ha molts altres articles en aquest àmbit que arriben a conclusions similars [ZSYL+22], on s'analitza també l'NO2 i el BC.

En el desenvolupament dels models, s'ha pogut comprovar com el perceptró multicapa és capaç d'obtenir resultats molt millors respecte a la regressió lineal. Aquest fet indica que aquests models no lineals poden trobar relacions més complexes i profundes, que permeten realitzar prediccions més encertades. En general, els models usats tenen molt millor rendiment en el train, però, tot i això, generalitzen bastant bé, i hem sigut capaços d'evitar i solucionar els problemes dels perceptrons multicapa amb cada iteració (under-over fitting sobretot, tot i que en les proves per la quarta iteració també vam trobar inestabilitat quan canviàvem la mida del batch i el learning rate, o convergència excessivament ràpida). Una variància explicada de 0.50 considerem que és bastant bona, sobretot si tenim en compte que el rendiment mental és una característica molt complicada de mesurar i determinar.

Cal tenir en compte, però, que els resultats del model lineal els hem pogut analitzar i determinar quines eren les variables amb més influència, mentre que els de l'MLP no, ja que és un model *black-box*.

Un fet a comentar és que els resultats, però, depenen bastant d'un factor aleatori. Per exemple, en l'últim model poden passar de 0.47 fins a 0.52, sense canviar cap paràmetre. Aquest fet ens mostra com en són de sensibles les xarxes neuronals a factors com la inicialització, que provoquen que trobin un mínim diferent. A més, veient l'evolució creiem que amb més iteracions els resultats amb el train millorarien, però no els del validation, ja que semblen quedar ja estancats.

En conclusió, hem observat com els perceptrons multicapa (models no lineals) són capaços de superar amb escreix els models lineals, valorant també que tenen un potencial més elevat malgrat la manca d'interpretabilitat. A més, hem pogut detectar la relació entre el rendiment mental i les variables esmentades com la contaminació, fet que indica que seria interessant investigar amb més profunditat aquesta relació amb un conjunt de dades més gran per veure com certs factors ambientals i de contaminació ens afecten en el nostre dia a dia. Amb tot, sembla que caldria fer un esforç per intentar reduir els nivells de contaminació, ja que hem pogut analitzar com afecten en el rendiment (i, per tant, la salut) mental.

Referències

[AAB+15] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[GB10] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.

[GRT+22a] Florence Gignac, Valeria Righi, Raul Toran, Lucía Paz Errandonea, Rodney Ortiz, Bas Mijling, Aytor Naranjo, Mark Nieuwenhuijsen, Javier Creus, and Xavier Basagaña. Cities-health barcelona panel study results. *Zenodo (CERN European Organization for Nuclear Research)*, 04 2022.

[GRT+22b] Florence Gignac, Valeria Righi, Raül Toran, Lucía Paz Errandonea, Rodney Ortiz, Bas Mijling, Aytor Naranjo, Mark Nieuwenhuijsen, Javier Creus, and Xavier Basagaña. Short-term no2 exposure and cognitive and mental health: A panel study based on a citizen science project in barcelona, spain. *Environment International*, 164:107284, 06 2022.

[Hub64] Peter J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35:73–101, 03 1964.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[ZSYL+22] Mohammad Javad Zare Sakhvidi, Jun Yang, Emeline Lequy, Jie Chen, Kees de Hoogh, Noémie Letellier, Marion Mortamais, Anna Ozguler, Danielle Vienneau, Marie Zins, Marcel Goldberg, Claudine Berr, and Bénédicte Jacquemin. Outdoor air pollution exposure and cognitive performance: findings from the enrolment phase of the constances cohort. *The Lancet Planetary Health*, 6:e219–e229, 03 2022.