Universitat Politècnica de Catalunya

Facultat d'Informàtica de Barcelona

Pràctica 1:
Perceptró Multicapa

Grau en Intel·ligència Artificial

Xarxes Neuronals i Deep Learning

Anna Casanovas Poirier - 48039079Z

Abril Risso Matas - 45182567X

07/05/2024

Índex

1 Introducció

L'objectiu d'aquesta pràctica és veure una aplicació de l'aprenentatge automàtic treballant amb un conjunt de dades, fent una anàlisi i un preprocessament d'aquestes. Seguidament, es fa un modelatge fent ús primer de models linears (logístic regression, random forest...) i després utilitzant una xarxa neuronal de Perceptró Multicapa. Finalment, es fa una comparació de tots els models per comprovar la seva eficàcia per a predir/classificar una variable.

2 Base de Dades triada

La base de dades que s'ha triat conté 10 anys del temps a diferents llocs d'Austràlia. Té variables que expliquen la temperatura, el vent, la pluja i més condicions meteorològiques. La variable objectiu és "RainTomorrow", aleshores l'objectiu és predir si l'endemà plourà o no a partir de la informació que tenim. Com es tracta d'una variable categòrica la tasca és de regressió logística.

3 Anàlisi Exploratòria de Dades

Abans de fer cap mena de preprocessament, s'ha fet una anàlisi de les dades. A continuació s'expliquen tots els passos que s'han fet, però es recomana consultar el Notebook per visualitzar tot el procés abans d'entrenar els models. Per dur a terme l'anàlisi estadística, s'han creat dues llistes (*var_num* conté totes les variables numèriques i *var_cat* conté totes les variables categòriques).

3.1 Anàlisi Univariant

Com s'ha pogut examinar, el conjunt de dades conté **16 variables numèriques**. S'ha realitzat una taula per cada variable per poder observar un resum estadístic de cada variable. En aquestes taules es troba la quantitat d'observacions no nul·les, la mitjana, la desviació estàndard, el valor mínim, els diferents percentils (25, 50 i 75) i el valor màxim de cada variable. A continuació, s'han dut a terme tots els histogrames corresponents a cada variable numèrica.

Les variables **Cloud9am** i **Cloud3pm** representen la distribució de la cobertura dels núvols a les 9 del matí i a les 3 de la tarda mesurada en octes.

La variable **Evaporation** ens mostra un gràfic esbiaixat cap a l'esquerra el que ens indica que majoritàriament a Austràlia trobem taxes d'evaporació baixes. Semblant a la variable **Rainfall** amb la majoria d'instàncies entre valors 0-20, també esbiaixada cap a l'esquerra.

Les variables **Humidity9am** i **Humidity3pm** mostren el nivell d'humitat respectiu a aquelles hores del dia. La gràfica d'Humidity9am sembla una mica esbiaixada cap a la dreta indicant que la majoria de valors a les 9 del matí són bastant alts, en canvi, podem observar que la distribució d'humitat a les 3 de la tarda s'assembla més a una distribució normal.

La variable **MinTemp** i **MaxTemp** indicant la distribució de les temperatures màximes i mínimes, sembla bastant simètrica i podria aproximar-se a una normal, la qual cosa indica que la majoria de dies

tenen temperatures que es concentren al voltant d'una temperatura mitjana.

Les variables **Pressure9am** i **Pressure3pm**, mostren les pressions atmosfèriques en les hores corresponents. Les dues distribucions semblen pràcticament normals i centrades al voltant d'un valor mitjà comú.

Les variables **Temp9am** i **Temp3pm** també semblen pràcticament normals encara que el valor promig a les 9am és de 16.89 i a les 3pm és de 20.82, indicat que a les 3 de la tarda fa més calor.

Les variables **WindGustSpeed**, **WindSpeed9am** i **WindSpeed3pm**, representen les velocitats del vent màximes, a les 9 del matí i a les 3 de la tarda respectivament. Com es pot observar, les 3 gràfiques semblen una mica esbiaixades cap a l'esquerra indicant que no hi ha un alt nombre de fortes ventades, tot i això, semblen tenir una distribució bastant normal.

3.2 Anàlisi Bivariant

Per fer l'anàlisi bivariant, s'ha realitzat els **barplots de les variables categòriques** juntament amb la variable objectiu RainTomorrow, que és una variable binària, i també s'ha realitzat els boxplots de les variables numèriques juntament amb la variable objectiu.

Per visualitzar les relacions entre les variables numèriques, s'ha dut a terme una **matriu de confusió**. D'aquesta manera hem pogut observar aquelles variables més correlacionades amb altres i aquelles que no tenen relacions tan fortes amb altres variables.

A continuació, per veure la rellevància de les variables categòriques respecte a la variable objectiu, s'ha realitzat un **anàlisi de redundància mitjançant Chi-Square**, ja que la nostra variable objectiu també és categòrica. Per tant, s'ha vist el p-valor resultant d'aplicar aquest test, i s'ha pogut veure que totes les variables tenen rellevància, amb un p-valor inferior a 0.05.

4 Estratègia de Preprocessament

Un cop ens hem familiaritzat amb les dades, s'ha procedit al seu preprocessament.

Recodificació de les variables: El primer que s'ha fet ha estat fer una recodificació d'algunes variables, ja que en alguns casos per les variables categòriques, hi trobàvem moltes categories. En el cas de la variable Date s'ha dividit en tres variables diferents: el dia, el mes i l'any de la pluja. Després la variable location, que ens indicava la ciutat d'Austràlia on havia plogut, s'ha passat a la regió que es troba aquesta ciutat i així hem passat de tenir unes 50 ciutats a 8 regions. Finalment, hem passat les variables que ens indicaven la direcció del vent a què només ens indiqui si anava cap al sud, nord, est o oest.

Tant per poder fer servir mètodes d'imputació i per realitzar el nostre model de regressió logística s'han de recodificar les nostres variables categòriques. Per executar aquesta tasca s'ha fet servir One-hot encoding, que crea noves variables binàries per cada categoria de les nostres variables categòriques.

Identificació i tractament d'outliers: Per identificar els outliers s'han fet una sèrie de boxplots on es poden veure els valors extrems. Com s'ha vist a l'apartat anterior, moltes de les nostres variables tenen una distribució normal, cosa que fa que poguéssim eliminar els outliers amb el mètode IQR. Tot i això, s'ha observat que a la majoria de les variables no trobem valors extremadament anormals, per tant, s'ha considerat mantenir la majoria d'aquests. Tot i això, s'ha eliminat els valors superiors a 250mm de la variable Rainfall, ja que aquests valors són extremadament estranys i podrien esbiaixar el model. També s'ha eliminat les files que contenen un nivell d'evaporació superior a 20, ja que són valors molt poc comuns a Austràlia i podrien resultar erronis. Finalment, també s'ha eliminat les files on la velocitat del vent és superior a 100km/h, tot i que aquestes ventades poden ser molt fortes en climes específics, valors per sobre d'aquest llindar podrien representar errors o valors molt poc comuns. Amb aquest tractament d'oultliers s'han eliminat 310 files del conjunt de dades, la qual cosa, en comparació amb el nombre de files inicials (67021), no és un canvi gaire significatiu respecte la mida del dataset.

Identificació i tractament de missing values: Després, s'ha fet la identificació de missing values. S'ha vist que hi ha una gran quantitat de valors faltants, amb diverses variables que tenen més d'un 40 per cent de missing values (30.000 mostres aprox). Primer de tot s'han eliminat les files que tenen valors faltants de la nostra variable objectiu, RainTomorrow, ja que si no sabem el valor real, no sabrem si el nostre model l'haurà predit correctament. A més, com que tenim moltes mostres, s'han esborrat les files d'aquella variable que té el percentatge més gran de missing data (en el nostre cas Sunshine, amb un 48% de missings). Un cop feta aquesta eliminació s'ha passat de tenir 67021 mostres a la meitat (33455). Tot i que és una gran reducció, el nostre dataset encara té una mida considerable. Ara com a màxim les nostres variables tenen un 10% de missing values, indicant que les files eliminades no només tenien missings en la columna Sunshine, sinó que també hi contenien missings en altres variables. A més amb la gràfica realitzada sobre els valors faltants de les variables, podem veure que moltes files on hi ha missings a la variable Sunshine, també en tenen en altres variables. Aquest 10% de mostres sí que serà imputat. S'ha provat de fer una imputació amb KNN i també amb MICE. S'han observat resultats similars amb els dos mètodes, però a causa de la simplicitat del primer s'ha triat imputar les nostres dades amb el KNN.

Balanceig de les dades: Com s'ha observat a l'anàlisi exploratòria de dades, la nostra variable objectiu està molt balancejada, per tant, no ha fet falta utilitzar cap mètode de oversampling o undersampling.

Normalització de les dades: Per millorar el rendiment del nostre model i assegurar-nos que totes les variables tenen la mateixa importància al nostre model és necessari fer una normalització de les nostres dades. Per dur a terme aquesta tasca s'han provat dos mètodes diferents: *MinMax* i *StandardScaler*. Dels dos mètodes, havent fet un cross validation en el nostre model de regressió logística, s'ha conclòs que el millor mètode a implementar era *MinMax* en comparació amb el resultat de normalitzar les dades amb *StandardScaler*.

5 Remostreig

Un cop ideada l'estratègia de preprocessament, s'ha fet la partició de les dades. S'ha decidit fer una partició de train i test on s'han dedicat un 30% de les dades al test. Per fer la tria d'hiperparàmetres es farà un cross validation (un GridSearch) a les dades del train.

6 Model Lineal Base

Com a model linear base s'ha triat la regressió logística. S'han fet dos models, el primer sense aplicar cap regularització i el segon afegint-li. En el cas del segon, s'ha fet un CrossValidation per trobar quina penalització utilitzar (L1 i L2) i quin paràmetre C, que controla la intensitat de la regularització per prevenir el overfitting. C és la inversa de la lambda (força de regularització). Finalment, els paràmetres triats amb la cross validation han estat: una regularització L1, que no només redueix la influència d'una variable sinó que també fa feature selection (Lasso). Després s'ha triat un C = 100. Això vol dir que la lambda és petita i, per tant, el poder de la regularització és petit.

Tot i que el model anterior no dona resultats dolents, també s'ha volgut entrenar un altre model per veure si canvia molt el rendiment. S'ha realitzat un RandomForest. S'ha fet una cerca d'uns entre molts hiperparàmetres que el model permet: 'n_estimators','max_depth'. Els hiperparàmetres triats amb el CrossValidation finalment han estat: 'max_depth' = 30 i 'n_estimators' = 300.

6.1 Resultats model base

A continuació es mostren els resultats obtinguts amb la Regressió logística sense la regularització i també amb la regularització, establint una llavor igual a 42:

Taula 1: Resultats regressió logística

Model Name	Accuracy	F1 Score	Recall	Precision
LinearRegression	0.803627	0.800365	0.793533	0.807315
LinearRegressionReg	0.804324	0.800528	0.791524	0.809739

Veiem que els dos models tenen un f1-score de més del 0.8, per tant, és un bon model. Trobem que tot i que les mètriques siguin lleugerament millors a l'afegir la regularització, els resultats no varien de manera exagerada. Això vol dir que el model sense la regularització ja no feia overfit. Tot i això, decidim quedar-nos el model LinearRegressionReg. En la matriu de confusió dels models podem veure que el model tendeix a predir més dies de no pluja a dies de pluja, per tant, hi ha més True Negatives que True Positives.

Finalment, veiem que després d'entrenar també el model RandomForest amb CrossValidation els resultats donen una mica millor que el Logístic Regression arribant a una f1-score de 0.817.

7 Procés iteratiu - Perceptró Multicapa (MLP)

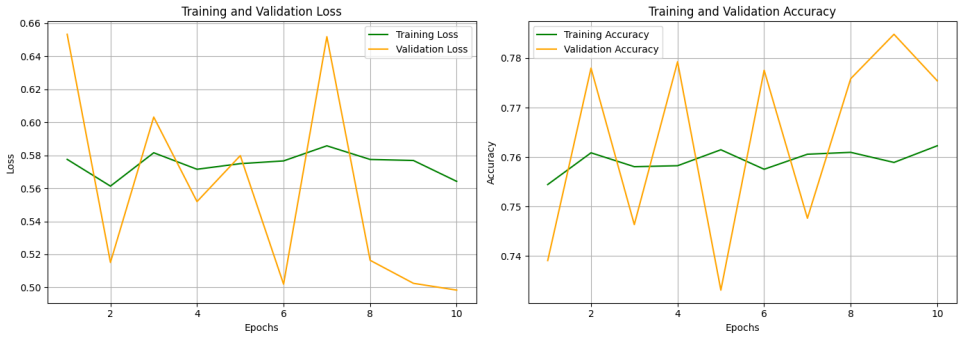
Un cop hem fet un model lineal base, el nostre objectiu és entrenar de manera iterativa un Perceptró Multicapa, que farà que el model millori. En aquest apartat es provaran diferents paràmetres per millorar el model: el optimizer (Adam, SGD), el learning rate, nombre de epochs, regularització (L1, L2 o cap), nombre de capes (entre 1 i 3), nombre de neurones per capa, la funció d'activació (relu, tanh...), la batch size i l'aplicació o no de dropout.

Perquè l'avaluació del model sigui més robusta i les nostres conclusions siguin més consistents, s'ha decidit executar cada model de perceptró multicapa 3 cops. Un cop executats, es farà la mitjana de les mètriques que avaluem.

7.1 Primer model

El primer model que s'ha fet és el més senzill de tots. Té solament una capa, amb dues neurones i una activació softmax. Té 10 èpoques i un learning rate bastant alt, de 0.1. L'optimitzador utilitzat en aquest cas és l'Adam (adaptive moment estimator).

A la següent Figura veiem l'evolució de la pèrdua i l'accuracy al llarg de les èpoques en el primer entrenament (dels 3 realitzats).



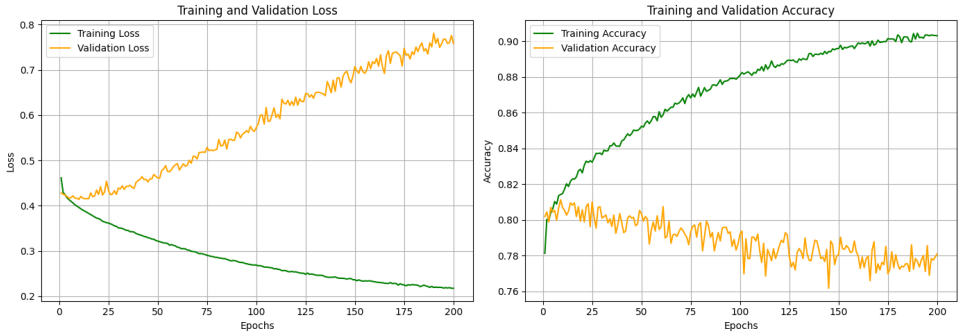
La loss de l'entrenament al llarg de les epochs va disminuint, cosa que és bona, ja que vol dir que el model està aprenent de les dades del train, però la loss de la validació mostra pics i molta variabilitat, indicant que igual no generalitza bé a noves dades. A més, les dades de validació no arriben a convergir mai. En el cas de la accuracy, per l'entrament és estable i millora de manera lleu però a la validació és molt variable i aquest comportament volàtil pot voler dir que el model no és estable.

Aquest model ha acabat obtenint els següents resultats mitjans de test: Test F1 Score de 0.749184 i una Loss de 0.9 en el primer entrenament, 0.47 en el segon i 0.49 en el tercer.

Aquests resultats ens diuen que el model ha aprés patrons, però encara hi ha un gran marge de millora, sobretot considerant la consistència del model entre les dades d'entrenament i de validació i la falta de convergència. La pèrdua al conjunt de proba és relativament baixa cosa que és positiva. Podem concluir que aquest model tot i ser molt senzill, ja fa una feina prou bona, però canviant altres paràmetres i afegint més capes encara hi ha marge de millora.

7.2 Segon model

Com hem vist que el model anterior feia underfitting, s'ha decidit fer un model amb tres capes. La primera amb 64 neurones, la segona amb 32 i la última amb dues neurones. També s'ha augmentat el nombre de èpoques i s'ha posat un learning rate molt més petit. Els resultats de la partició del test són: Loss d'entre 0.7 i 0.8 en els 3 experiments i F1-score de 0.770079

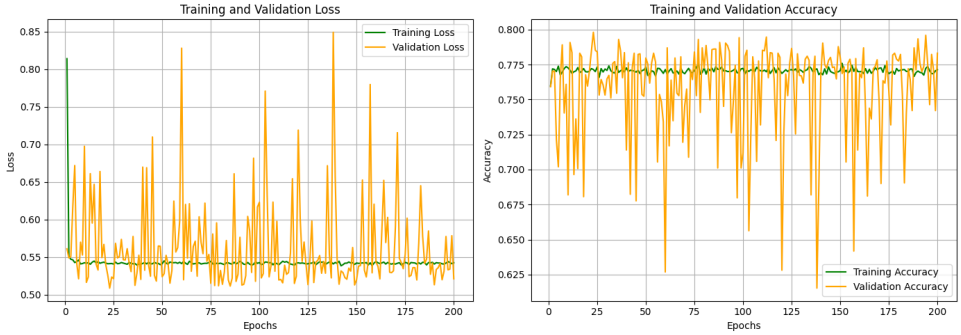


Sorprenentment, aquest model és pitjor que l'inicial. Dona un f1-score més petit i a més la pèrdua és molt més alta. Com es pot veure a la figura, aquest model fa molt overfitting, ja que la accuracy del train arriba fins a més de 0.9 però la loss al validation, en comptes de reduir-se com al train, augmenta de manera significativa. Veiem que el conjunt del train convergeix però la pèrdua de la validació augmenta de manera linear.

Aquest overfitting pot ser degut a que s'han afegit tres capes, i una manera de solucionar-ho pot ser afegint una regularització al model i dropout (eliminació d'algunes neurones de manera aleatòria per a la millor generalització del model).

7.3 Tercer model

Per fer el tercer model, s'han utilitzat tres capes com abans, però se li ha afegit una regularització de l1 = 0.001 i l2 = 0.002 en les capes ocultes.

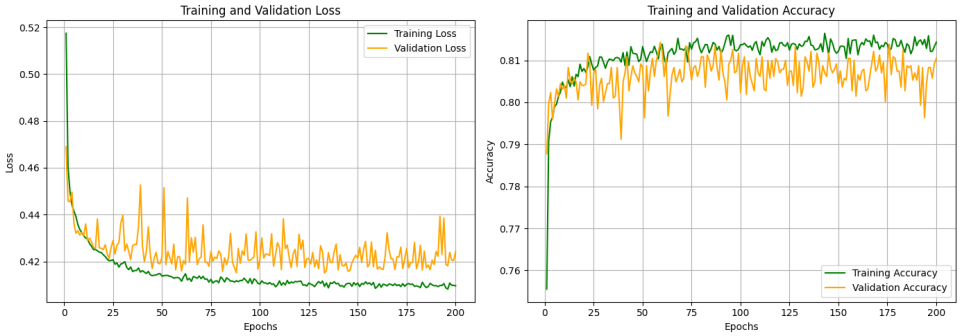


Aquest model té un Loss d'entre 0.51 i 0.53 en els 3 experiments i un F1-Score de 0.782039 mitjà. Té uns millors resultats per la partició del test que els dos anteriors. El problema és que, com es pot veure a la figura, per la partició del validation és molt irregular per totes les èpoques, no arriba a convergir. Tot i que aquest model és una millora, encara pot anar a millor.

Per això, també se li ha afegit més tard un dropout de neurones del 0.01. Això fa que durant el procés es vagin eliminant conexions i per tant el model encara generalitzi més. Aquest dona millors resultats arribant a un 0.81 de les mètriques avaluades.

7.4 Model Guanyador

Finalment, després d'haber provat amb paràmetres i tamanys diferents, s'ha arribat a un model que ni fa infraajust ni sobreajust i generalitza molt bé. El model final és com el últim que s'ha fet, utilitzant la regularització l2 i el dropout, però en comptes d'utilitzar tres capes, n'utilitza dues. Hem trobat que com més senzill és el model, millors resultats dona i més fàcil és que no faci overfitting de les dades.



A la figura veiem que les dades del validate tenen la mateixa tendència que les del train, on la Accuracy va augmentant i la Loss disminueix al llarg de totes les èpoques. La Loss de la partició del test és de 0.41, la més baixa vista fins ara. El f1-score d'aquesta partició de mitjana en els 3 experiments és de 0.813188, superant al nostre model base de Logistic Regression.

8 Conclusions

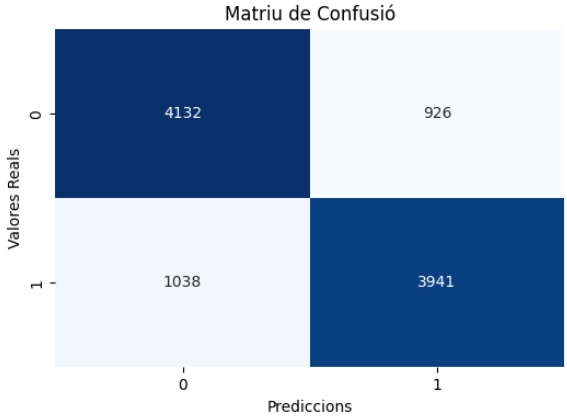
A continuació es proporciona una taula comparativa amb tots els models que s'han entrenat durant la pràctica i també una anàlisi comparativa entre el model lineal base i el nostre perceptró multicapa final.

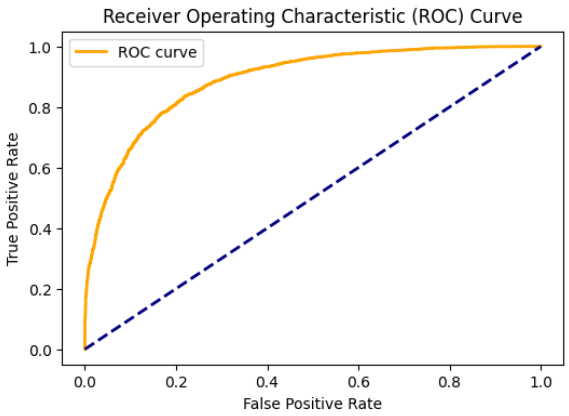
Taula 2: Resultats comparatius de tots els models

Model Name	Accuracy	F1 Score	Recall	Precision
LinearRegression	0.803627	0.800528	0.791524	0.809739
LinearRegressionReg	0.804324	0.800569	0.791725	0.809612
MLP 1	0.756302	0.749184	0.755678	0.779294
MLP 2	0.770117	0.770079	0.770082	0.770174
MLP 3	0.780811	0.780038	0.781326	0.785878
MLP 3 (dropout)	0.810435	0.810415	0.810545	0.810746
MLP FINAL	0.813291	0.813188	0.813197	0.813692

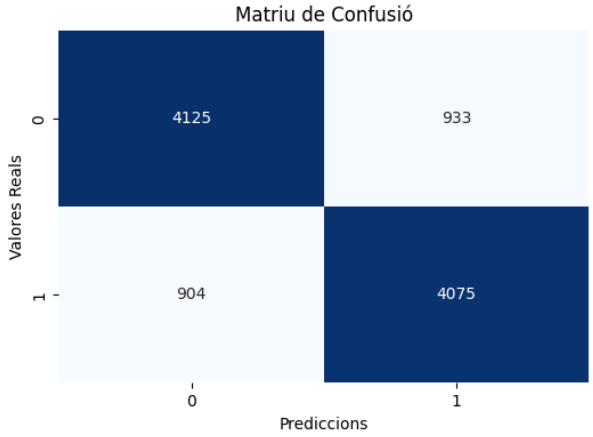
En el cas de la regressió logística veiem a la matriu de confusió que hi ha un nombre relativament alt de positius verdaders i negatius verdaders. El model detecta 926 instàncies com que plourà però és mentira i no plourà i 1038 com a que no plourà i realment plou. El model per tant s'equivoca més quan diu que

no plourà i realment sí ho farà. També podem visualitzar la corva roc. Com més aprop estigui la la corva taronja del borde superior esquerra, vol dir que el model té millor capacitat epr distingir entre les dues classes a diferents humbrals.





A continuació observem la matriu de confusió del model Perceptró Multicapa final en el millor dels 3 entrenaments.



Veiem que aquest té un nombre lleugerament major de mostres ben classificades. Ara ja no hi ha tanta diferència entre el nombre de mostres ben classificades en cada classe, en aquest cas el model s'equivoca pràcticament igual per les dues.

Després d'haver fet la comparativa dels models, podem concluir que amb la Regressió Lineal ja vam entrenar un bon model, arribant gairebé a la mateixa precisió que el nostre Perceptró Multicapa. En el cas d'aquest últim, hem pogut veure que el seu rendiment depén molt dels paràmetres que es configuren. Hem comprovat la importància d'utilitzar tècniques per regularitzar perquè el model generalitzi millor. El que ha costat més de l'experimentació amb el MLP ha estat trobar una combinació de paràmetres i capes que no faci overfitting.

Finalment, el nostre Perceptró Multicapa final supera en rendiment a la regressió logística. Aquest model és molt útil per a qualsevol aplicació real que tingui a veure amb la predicció del temps.