Universitat Autònoma de Barcelona

Anàlisi i predicció dels accidents de trànsit de Barcelona amb l'ús d'eines de Machine Learning

Alumne:	Jordi Solé Casaramona
Grau:	Empresa i Tecnologia
Tutora:	Gloria Estapé Dubreuil
Data de lliurament:	7 de juny de 2019

Abstract

According to the World Health Organization (2018), traffic crashes cost most countries 3% of their gross domestic product. Thus, both money and more importantly lives could be saved by any measure directed to preventing crash accidents. In particular, the development of predictors for traffic accidents could be used for a better public and more safe transportation.

This project aims to solve three different problems related to vehicle accidents in the city of Barcelona. It uses mainly raw data from traffic crashes provided by their local police, Guardia Urbana. The problems are, first, to predict the type of injuries produced in a given crash; second, to predict the model of a runaway car after an accident, taking into account the 200th most common cars in Barcelona; and third, predict which type of collision will occur. Related data from weather conditions, solar position, number and distance from traffic lights, and population density had also been used.

In this fast-paced era of information, there is so much data that humans alone cannot understand nor see the hidden patterns this valuable Big Data information has. Therefore, to deal with the three above mentioned problems, two different Machine Learning techniques had been used: Random Forest and Artificial Neural Networks algorithms. Furthermore, a comparison of the result of these two models has been performed, to evaluate which of the classifiers had a better fit in each problem. Different techniques were also used in order to solve severe class unbalance, binary multiclassification, sparse data and other problems that the raw data from Open Barcelona had.

The results show a good performance specially of the Artificial Neural Network algorithm, both for the prediction of the model of a runaway car and for the prediction of the type of collision. Indeed, the obtained accuracy has proved to be 42,9% and 54,8%, respectively, more accurate than the baseline model.

Índex

1. Introducció

1.1 Visió general

Aquest treball comença a partir de l'ambició d'aprofundir el coneixement sobre Python i el camp del Machine Learning, especialment sobre les Xarxes Neuronals i el Deep Learning. A més de ser aquesta una de les àrees on vull enfocar el meu futur professional.

En aquest estudi s'ha volgut analitzar i predir els accidents de trànsit de la ciutat Barcelona. Per fer-ho, s'han utilitzat dades de *Open Barcelona*, portal de l'ajuntament de la ciutat que té diferents bases de dades públiques. En aquest cas s'han usat les dades sobre accidents que compila la Guardia Urbana, i que es publiquen en tres blocs diferents. Cal tenir en compte que degut a la protecció de dades aquest tres datasets no es poden ajuntar per falta d'una clau primària com la matrícula del cotxe o el DNI de les persones.

Aquest treball consisteix en l'anàlisi d'aquestes tres bases de dades per separat, estudiant tres problemes diferents, que es descriuen a continuació:

- La predicció de la **gravetat dels ferits** en els accidents: lleu, greu o mort.
- En el cas que un dels vehicles que intervenen en una col·lisió es doni a la fuga, **determinar el seu model** i així ajudar a la policia a reduir el ventall de possibilitats.
- Predir com serà el **tipus de col·lisió** que patiran els vehicles accidentats: frontal, lateral, etc.

Es disposa també de informació addicional, incloent la localització de l'accident (barri, carrer i districte), data i hora en el que es va produir i la resta de vehicles implicats en aquell mateix accident. A més, s'han considerat als datasets totes les dades que poden influir en un accident com són: dades meteorològiques (precipitació, humitat, temperatura i vent), dades de la densitat de població de la zona, dades de sortides i postes de sol i finalment la localització dels semàfors. El procediment seguit per la incorporació d'aquestes dades s'explicarà més en detall en els següents apartats.

Al treball s'expliquen també tot els problemes que tenien aquestes dades, deguts a que no han estat processades prèviament. D'aquesta manera s'exposaran els procediments d'anàlisi de grans volums de dades, des de l'extracció fins l'anàlisi d'aquestes. La principal dificultat trobada està relacionada amb el problema dels ferits, ja que es tracta d'una base de dades descompensades *(Class Unbalance)* al haver-hi moltes més instàncies de ferits lleus (98%) que de la resta de classes. Chawla, Bowyer, Hall, & Kegelmeyer (2002) van ser dels primers en desenvolupar la tècnica de SMOTE *(Systhemic Minority Over-sampling Technique)* usada en aquest treball per resoldre el problema de les classes descompensades.

S'han utilitzat eines de Business Intelligence, i en particular el programari Tableau, per analitzar els resultats d'aquest estudi i determinar les possibles relacions entre els diferents atributs dels accidents estudiats. Finalment, s'han fet models de predicció usant algoritmes de Machine Learning (Random Forest i Xarxes Neuronals Profundes) usant el llenguatge de programació de Python amb llibreries com Sklearn i Keras. Ambdós tipus

d'algoritmes han sigut comparats per cada tipus de problema per veure quin obtenia millors resultats.

Els resultats principals del treball es poden resumir en : (1) Aconseguir una millora de *l'Accuracy* del model de gravetat dels ferits, encara que lleugera, combatent el desbalancejament de les classes. (2) L'obtenció d'un predictor de models de cotxe amb l'ús de Xarxes Neuronals, en el cas d'accident amb fuga d'un dels vehicles implicats, que aconsegueix una Accuracy un 42,9% millor que el classificador base. (3) L'obtenció d'un model que exhibeix una *Accuracy* un 54,8% millor que el classificador base en la predicció del problema del tipus de col·lisió.

1.2 Treballs relacionats

Diversos treballs previs han ajudat a enfocar aquest treball. La majoria. però, estan centrats en predir la gravetat dels ferits dels accidents de trànsit i el lloc on succeeix l'accident, a partir de l'ús de dades viaries molt precises.

L'estudi de Yuan, Zhou, Yang, Tamerius, & Mantilla (2017), on han superat el problema de *Class Unbalance* afegint exemples negatius, utilitza dades d'accidents entre 2006 i 2013 amb algoritmes com Random Forest i Xarxes Neuronals. Algoritmes que consideren que són els més efectius en problemes de classificació. Aquest treball també usa moltes fonts d'informació que afecten als vehicles a la carretera com són les condicions climàtiques, sortides i postes de sol, etc. I demostren que aquesta informació addicional és positiva per augmentar la precisió dels algoritmes.

Un altre estudi en el que es tracta el problema de les classes descompensades és el de Pozzolo, Caelen, & Johnson (2019), on s'exposa codi Python i eines com SMOTE per resoldre'l pel cas de detecció de frau bancari.

Un altre treball molt revelador i que ha sigut de gran ajuda ha estat Wilson (2018), en el que s'inclou codi Python per visualitzacions dades geoespacials.

Theofilatos, Yannis, Kopelias, & Papadimitriou (2016) troben una relació negativa entre la velocitat i els accidents produïts en autopistes. Finalment l'estudi de Tamerius, Zhou, Mantilla, & Greenfield-Huitt (2016) conclou que, per l'estat d'Iowa, la probabilitat d'accidents és major a la tarda que al matí, i que hi ha una possible relació entre la precipitació i els accidents en autopistes.

Fins on jo se, ningú no ha fet un algoritme per preveure el tipus de model en cas de fuga després d'una col·lisió, cosa molt útil per la policia. Tampoc s'ha trobat cap publicació on es predigui com serà la col·lisió dels vehicles implicats. I per últim, en cap de les publicacions consultades es tenen en compte altres vehicles que no siguin cotxes, com per exemple bicicletes, motocicletes, etc.

Els algoritmes utilitzats en aquest treball estan basats en el Random Forest de l'estudi de Breiman (2001) i les Xarxes Neuronal Artificials basades en el Perceptró multicapa del treball de Gardner & Dorling (1998).

1.3 Fonts d'informació

En aquest apartat s'explica com s'han obtingut els dataframes finals per la fase d'anàlisi i predicció. Aquesta feina ha significat més del 70% de la càrrega de treball d'aquest estudi.

Els datasets trets de la base de dades de la Guardia Urbana tenen en comú un conjunt d'atributs que es detallen a la *taula 1*.

Taula 1: Atributs comuns en totes les bases de dades de la Guardia Urbana d'Open Barcelona.

Codi de l'accident	Codi districte	Nom districte
Codi barri	Nom barri	Codi carrer
Nom carrer	Dia (numero)	Nom dia de la setmana
Descripció dia (laboral/festiu)	Hora (format 24h)	Mes
Any	Latitud	Longitud

A més a més, cada dataset conté dades úniques que són les que s'expliquen a continuació.

Accidents i ferits:

Conté la informació relativa als accidents i els ferits en aquests successos. A més s'inclouen les dades del nombre de morts, ferits lleus i greus per cada codi d'accident, a més del nombre de vehicles implicats en cada col·lisió. Aquest Dataframe compta inicialment 24.000 entrades corresponent als anys 2016 i 2017.

Dades del portal *Open Data Barcelona*.

Models i Marca dels vehicles:

Inclou base de dades hi ha totes les dades comunes i altres dades en relació al tipus de vehicle involucrat en l'accident. Aquests atributs són la descripció del model, descripció de la marca, descripció del tipus de vehicle, descripció del color i antiguitat del carnet. A més de si hi ha hagut un vianant involucrat a l'accident. Aquestes dataframe conté dades de 2016 a 2018 amb més de 58.000 entrades inicialment.

Dades del portal *Open Data Barcelona*.

Tipus de col·lisió:

Per aquesta base de dades es troba la descripció del tipus de col·lisió produïda a l'accident (Col·lisió frontal, lateral, etc.). Les dades usades són de 2016 i 2017 per més de 20.000 registres inicialment.

Dades del portal *Open Data Barcelona*.

Meteorologia:

També es van afegir les dades meteorològiques de la zona al considerar-se la possibilitat que les condicions climàtiques anessin lligades amb els accidents com suggeria Tamerius et al. (2016). Aquestes dades inclouen la temperatura, humitat relativa, vent i precipitació.

La informació va ser obtinguda del *Meteocat* amb freqüència de cada 30 minuts per les 4 principals estacions meteorològiques de la ciutat durant els anys 2015 a 2017.

Localització dels semàfors:

Ubicació amb latitud i longitud dels semàfors de la ciutat de Barcelona. Informació que es va considerar rellevant ja que aquests es situen en moltes ocasions entre interseccions, zones propenses a accidents per poca visibilitat. És a dir, la ubicació dels semàfors ens servirà per determinar on són les interseccions entre les vies.

Dades del portal *Open Data Barcelona*.

Sortides i Postes de Sol:

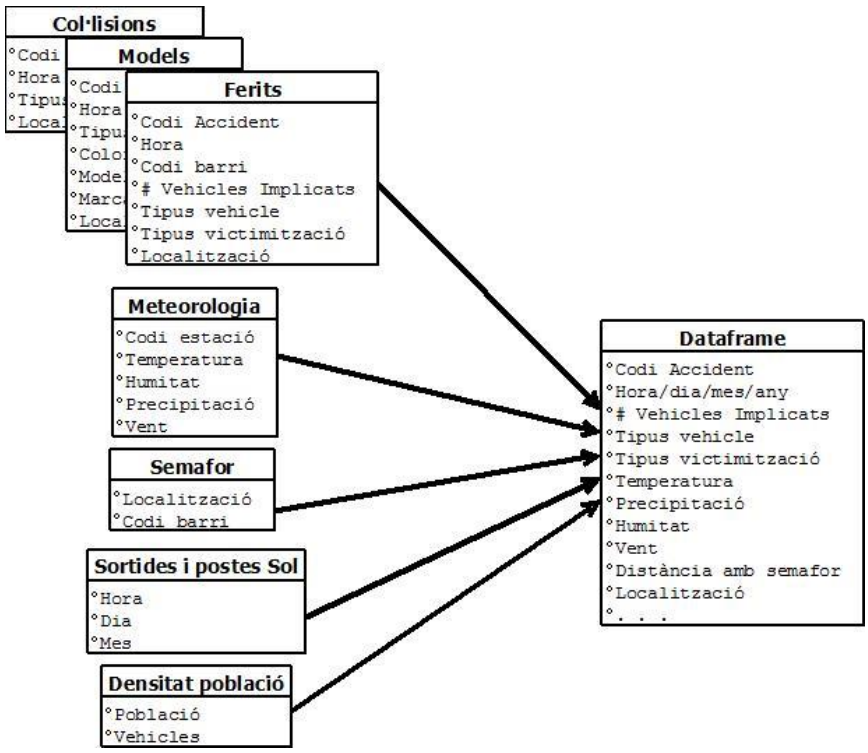
Les postes i sortides de Sol també són un dels factors clau en els accidents, ja que el Sol pot enlluernar als conductors. Per tant, es vol veure si al voltant d'aquestes hores hi ha més accidents a la ciutat.

Les dades poden ser trobades a diverses webs com *salidaypuestadelsol.com* i seran usades dades de 2015 a 2018.

Densitat de població i de parc automobilístic:

Es consideren també les variables de densitat de població i de vehicles per cada barri. Es vol relacionar si en barris amb més població i densitat de trànsit hi ha més accidents. Aquestes dades es troben també a la web de *l'Ajuntament de Barcelona*.

A la il·lustració 1 s'exposa un petit esquema per visualitzar els components del Dataframe de ferits per aquest treball.



Il·lustració 1: Simplificació de les dades que integren el Dataframe analitzat. En aquest cas només pel problema de la predicció dels ferits.

2. Procés d'anàlisi de dades

2.1 Data Understanding

En aquest apartat s'explica com s'ha realitzat una primera anàlisi de les dades per poder:

- Determinar els valors acceptables per cada atribut.
- Seleccionar els valors significatius.
- I finalment, visualitzar geoespacialment els accidents.

Aquesta primera anàlisi permet veure cap on enfocar el treball, quins són els outliers, dades a netejar i possibles relacions entre variables. Un cop identificats el problemes que tenen els datasets, a *l'apartat 2.2* tractarem aquestes dades.

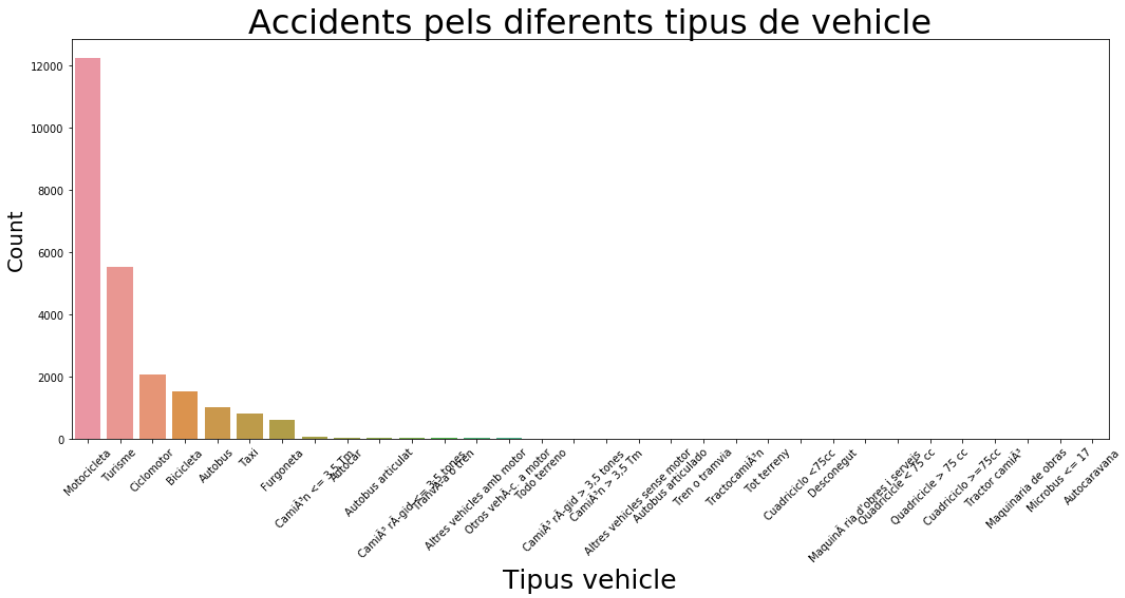
En primer lloc, s'observa l'aspecte inicial del dataframe, *il·lustració 2*. S'ha de tenir la cura de fer un procés de neteja de dades acurat per tenir així valors raonables per cada un dels atributs. Per exemple no es pot tenir, al camp de dia, un valor de 32 o tenir un codi de districte negatiu.

	NÃºmero_d'expedient	Codi_districte	Nom districte	Codi_barri	Nom_barri	Codi_carrer	Nom carrer	DescripciÃ³_dia_setmana	DescripciÃ³_victimitzaciÃ³	Dia de mes
0	2017S000001	5	SarriÃ - Sant Gervasi	26	Sant Gervasi - Galvany	187105	Madrazo	Diumenge	Ferit lleu	1
1	2017S000002	2	Eixample	5	el Fort Pienc	197302	Marina	Diumenge	Ferit lleu	1
2	2017S000004	2	Eixample	6	la Sagrada FamÃlia	89004	Consell de Cent	Diumenge	Ferit lleu	1
3	2017S000005	10	Sant MartÃ	66	el Parc i la Llacuna del Poblenou	243206	Pamplona	Diumenge	Ferit lleu	1
4	2017S000005	10	Sant MartÃ	66	el Parc i la Llacuna del Poblenou	243206	Pamplona	Diumenge	Ferit lleu	1
5	2017S000006	1	Ciutat Vella	1	el Raval	352100	Valldonzella	Diumenge	Ferit lleu	1

Il·lustració 2: Dataframe de ferits abans de fer cap modificació.

Per l'atribut de Descripció victimitzacions, a més de tenir la capçalera corrompuda, deixa entreveure un dels principals problemes per aquest dataset que és el gran nombre de ferits lleus respecte a les altres classes.

Per la selecció dels valors significatius s'ha fet ús de diferent llibreries (*Seaborn, Pyplot* o *Folium*) per mostrar les dades i veure a primera vista si hi ha valors que degut a que són poc significatius, es poden deixar fora de l'anàlisi sense que aquest perdi massa variància de la informació.

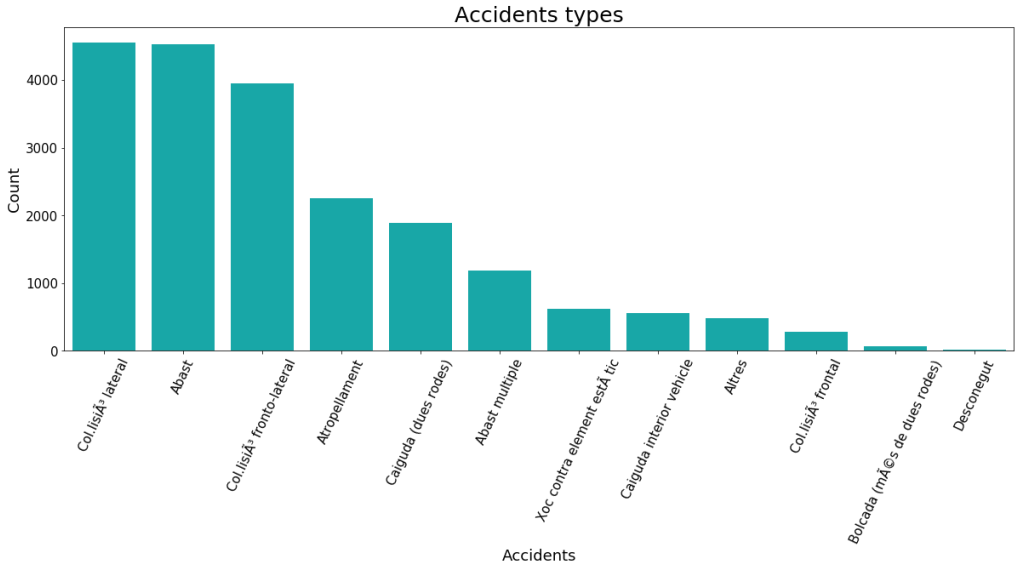


Il·lustració 3: Gràfica amb el nombre d'accidents per cada tipus de vehicle dels anys 2016 i 2017.

Per exemple, i pel que fa als accidents per tipus de vehicle (*il·lustració 3*), les dades mostren que la majoria d'accidents de la ciutat de Barcelona són amb motocicletes, amb més 12.000 entrades. A una distància considerable venen els turismes, seguit dels ciclomotors i de les bicicletes, dos vehicles més de dues rodes.

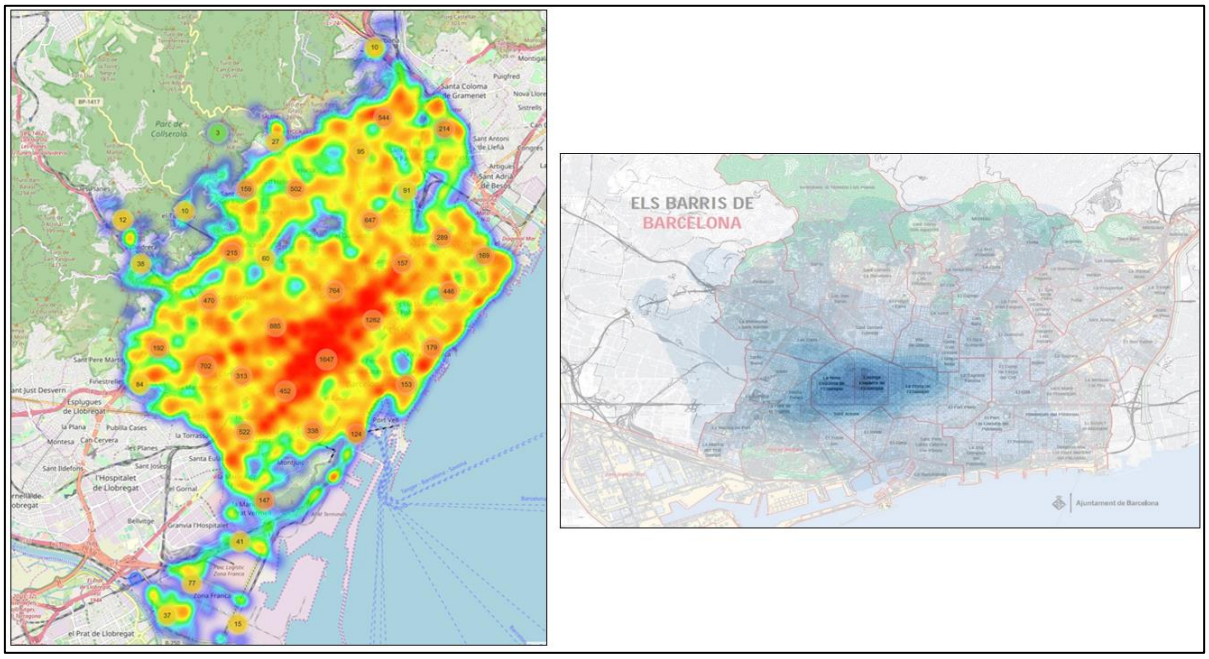
Ja que la majoria d'accident es concentren a uns pocs tipus de vehicles, la nostre anàlisi es limitarà als 7 vehicles amb més accidents. És a dir, des de motocicletes a furgonetes, deixant fora altres vehicles com autocaravanes o trens al comportar una part ínfima dels accidents.

Una altra mostra de dades a netejar dels datasets són les instàncies marcades com a Desconegut, per exemple als tipus de col·lisió, *il·lustració 4*. Per aquest motiu, aquest tipus d'entrades que es consideren que no aporten informació rellevant seran descartades del dataframe per reduir així el nombre de possibles classes del problema.



Il·lustració 4: Tipus de col·lisions als accidents de la ciutat de Barcelona.

Per últim, s'ha utilitzat el *heatmap* de la *il·lustració 5* per saber on hi ha un major nombre d'accidents a la ciutat.



Il·lustració 5: Heatmaps de les àrees amb més accidents a la ciutat de Barcelona.

Clarament es veu que la zona amb més accidents de la capital catalana és la zona de l'Eixample i la Diagonal, marcada en vermell degut a la gran quantitat d'aquests. Altres zones amb menys accidents són per exemple la serra de Collserola o Montjuic, zones menys transitades marcades amb verd.

2.2 Data Cleaning

Aquest és el nom amb el que es coneix el procés de neteja de dades i el seu tractament perquè es puguin acabar incorporant als nostres algoritmes de Machine Learning.

Com s'ha pogut veure abans a la *il·lustració 2*, hi ha valors corromputs o erronis que s'han de netejar per tal de que les dades siguin el més correctes possibles per aconseguir així un bon resultat amb la fase de l'anàlisi i predicció amb eines de Machine Learning.

2.2.1 Unió de les dades

Primerament ha estat necessari corregir les capçaleres dels atributs ja que per fer la unió de les dades *(Joins)* entre diferents dades, Python necessita que els atributs tinguin el mateix nom a les dues bases de dades en que es realitzarà aquesta operació.

Com mostra la *il·lustració 1*, hi ha moltes fonts de dades utilitzades per obtenir el *Dataframe* per a cada problema. La majoria d'elles s'han fet amb la intersecció dels dos conjunts seguint atributs com el dia, hora, més i any de l'accident o bé per la localització precisa de l'accident.

Cal reiterar el problema que es genera al voler ajuntar els *Dataframe* dels tres diferents problemes esmentats anteriorment. Ja que sí que aquests tenen un mateix codi d'accident però per exemple és impossible de relacionar els ferits amb el model dels vehicles. Això és degut a que per les dades de ferits només tenim el tipus de vehicle (cotxe, motocicleta, etc.). Si un codi d'accident té 2 vehicles implicats serà impossible saber quin ferit era a cada cotxe ja que no es disposa de la matricula del cotxe en que anava cada ocupant.

2.2.2 Dades corrompudes o errònies

Les dades corrompudes són aquelles que degut als accents o per altres motius agafen formats estranys com els vistos a la *il·lustració 2*. Aquestes han sigut corregides amb codi per tornar-les al seu estat original.

Pel que fa a les dades errònies, el que es va fer és a partir de gràfics o mirant els valors únics per cada variable, veure quins valors no tenien sentit. És a dir, detectar els outliers i dades mal classificades.

Un exemple és al dataset dels models de cotxe, és la *il·lustració 6*, on es veu com la marca BMW s'ha trobat escrita com a B M W, B.M.W o BMW. Aquest tipus de correccions s'han detectat fent un *distinct* sobre els noms dels models i després reclassificant canviant el nom de les entrades errònies, aconseguint així la coherència d'atribuir un sol nom a la marca. D'aquesta manera, si es filtra per BMW apareixeran totes les instàncies i no perdrem cap dada per estar mal escrita.

```
### Manualy fix the different names the police has given to the same brand.
accidents_vehicles = accidents_vehicles.replace({'A.U.D.I.': 'AUDI'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'A U D I': 'AUDI'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'MERCEDES-BENZ': 'MERCEDES'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'MECEDES': 'MERCEDES'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'MERCEDES B': 'MERCEDES'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'B.M.W.': 'BMW'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'B M W': 'BMW'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'MICROCAR': 'SMART'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'JAGUAR LAND ROV': 'LAND ROVER'}, regex=True)
accidents_vehicles = accidents_vehicles.replace({'Turismo': 'Turisme'}, regex=True)
```

Il·lustració 6: Exemple del tractament de dades errònies.

També es va determinar el rang de valor possibles que poden tenir els diferents atributs del *dataframe* per prescindir dels registres que contenen valors fora d'aquest rang. Es descarten per exemple, entrades que tinguin un codi de barri negatiu o major als 73 barris de la ciutat de Barcelona. Això es va fer per cada atribut, fins assegurar-nos amb mètriques com mitjanes, modes, i desviacions estàndard que no hi havia valors estranys a les nostres dades.

2.2.3 Dades en blanc

Els registres que contenien dades en blanc han sigut esborrades del *dataframe* ja que si per exemple no s'especifica el nombre de vehicles (és desconegut o nul) no es pot tractar correctament aquesta instància. Aquest procediment es fa per no incorporar valors erronis al nostre model.

La tècnica d'omplir els valors *null* amb la mitjana de la mostra o un valor aproximat es va desestimar degut a la gran quantitat de dades dels *datasets*.

2.2.4 Dades duplicades

Les dades duplicades són un problema i difícils de localitzar amb més de 24.000 entrades. Són dades que no són errònies i no els hi falta cap camp, però que a l'hora de fer l'anàlisi farà que el valor duplicat agafi més pes respecte la resta. Per evitar que es produís aquest biaix, es va fer ús d'eines de Python per detectar aquests valors i eliminar-los deixant només una de les entrada.

2.2.5 Selecció d'atributs

Pel dataframe s'agafen tots els atributs de caràcter numèric, ja que són els més fàcilment interpretables pels algoritmes. També es deixen variables no numèriques que seran processades per transformar-les en numèriques.

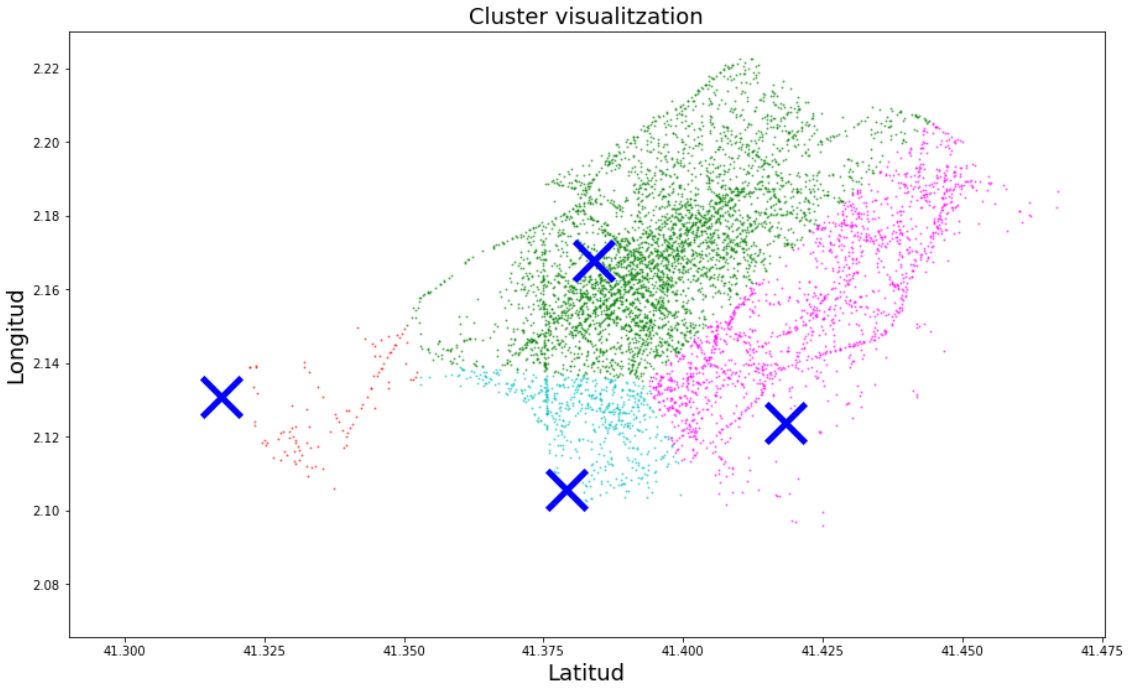
Per exemple, el nom del carrer no va ser passat al *dataframe* final, però sí el seu codi, ja que una dada numèrica ens dona molt més valor per l'algoritme que un nom d'un carrer. Això és degut a que una *string* no pot ser processada pels algoritmes de Machine Learning i que, a més com hem vist anteriorment, pot estar escrit de diferents formes. Per tant, els codis únics són per norma general més útils i precisos.

2.2.6 Atributs calculats

A continuació i un cop acabada tota la neteja i selecció de dades, s'han afegit atributs que resultin de diferents càlculs a partir dels *dataset* complementaris i que poden ajudar a millorar la precisió dels algoritmes.

El primer és el de la distància entre la localització de l'accident i del semàfor més proper. Aquesta dada permetrà saber si l'accident va tenir lloc prop d'interseccions. Per calcular aquest nou atribut, es va crear un algoritme que agafava la localització de l'accident i calculava la distància a tots els semàfors. Per fer aquest càlcul més ràpid, només es comparava amb semàfors que tinguessin el mateix codi de barri que l'accident, reduint per tant el nombre de distàncies que aquest algoritme havia de comparar. Un cop comparats tots els semàfors s'agafava la distància més petita i s'afegia com a nou camp per aquell codi d'accident.

El següent càlcul que es va realitzar va ser assignar a cada accident l'estació meteorològica més pròxima per tenir així les dades meteorològiques ajustades a les diferents zones de la ciutat. Aquesta feina es va realitzar usant un algoritme semblant a l'anterior modificat per que produís 4 clústers diferents entorn a les estacions meteorològiques més pròximes. El resultat d'aquest el veiem a la *il·lustració 7*.



Il·lustració 7: Clustering amb color els diferents accidents de Barcelona i les 4 estacions meteorològiques.

Finalment, també es va calcular la diferència entre l'hora de l'accident i la sortida o posta de Sol fent ús d'un bucle for. I. com la resta de resultats, aquests es van afegir al *dataframe* com a un nou atribut relacionat amb cada accident. A més, també es va agafar el mínim d'aquests per veure si realment la posició del Sol respecte l'horitzó té relació amb l'accident. D'aquesta manera es tenien les dades preparades pel següent pas: el processament de la informació obtinguda.

2.3 Data Processing

Aquest processament de dades només ha sigut realitzat dataframes que anaven als algoritmes de Machine Learning, mentre que els dataframes originals es deixaven sense processar per l'apartat d'anàlisi amb Tableau.

2.3.1 One-Hot Encoding

En aquest pas, es va passar les dades no numèriques a numèriques, fent servir la tècnica de binarització *One-hot encoding*.

Aquest procés es va portar a terme per cada variable no numèrica seleccionada, com per exemple el tipus de vehicle: cotxe, motocicleta, bicicleta, etc. D'aquesta manera aconseguim dades binaries de dades no numèriques, útils sobretot en tasques de classificació. La *il·lustració 8* mostra un exemple del resultat pels atributs de vehicles.

Edat	Codi_barri	Codi_carrer	Hora_de_dia	Mes_de_any	Dia_de_mes	Autobus	Bicicleta	Ciclomotor	Furgoneta	Motocicleta	Taxi	Turisme
50	26	187105	4	1	1	0	0	0	0	0	0	1
27	6	89004	7	1	1	0	0	0	0	1	0	0
34	6	89004	3	8	3	0	0	0	0	0	1	0
40	66	243206	7	1	1	0	0	1	0	0	0	0
37	66	243206	7	1	1	0	0	1	0	0	0	0
35	18	270209	14	1	1	0	0	1	0	0	0	0
50	18	270209	16	1	1	0	0	0	0	1	0	0
24	18	270209	22	3	24	0	0	0	0	0	1	0
47	18	270209	22	3	24	0	0	0	0	0	1	0
29	67	700662	15	1	1	0	1	0	0	0	0	0
23	5	100800	10	1	1	0	0	1	0	0	0	0

Il·lustració 8: Dataframe de ferits després d'aplicar-hi la tècnica de binarització One-hot encoding.

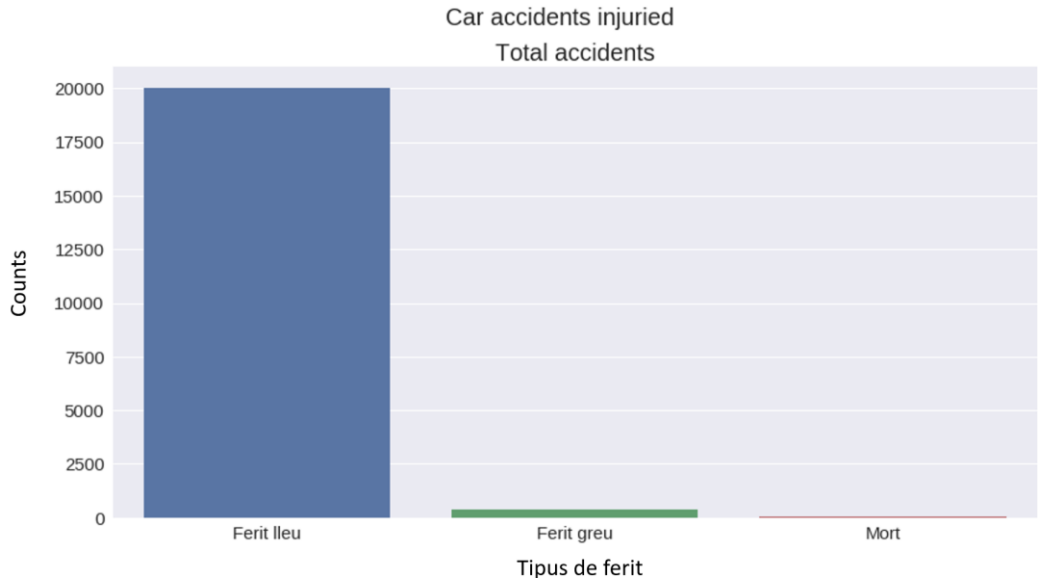
2.3.2 Normalització

Un procés clau a l'hora de tenir unes dades més interpretables pel algoritmes de Machine Learning és la seva normalització, aplicant a les dades unes transformacions que tinguin en compte el seu rang.

Aquesta transformació es realitza per així tenir les dades en un rang més petit i aquestes continguin la informació sobre la mitjana i la desviació estàndard de la mostra. Es va triar la tècnica del *Z-Score* per efectuar aquesta normalització. Així tots els atributs tindran rangs semblants i de cara als algoritmes això és molt important ja que sinó aquests donaran més pes a les variables amb valors més grans que amb petits. I així per tant, s'evita un biaix a les dades degut al pes d'aquestes *(weight bias)* sobre el model.

2.3.3 Mostres descompensades (Unbalanced data)

Un del problemes més importants a l'hora de fer prediccions amb targets com la gravetat dels ferits va ser descobrir que la mostra estava molt descompensada.



Il·lustració 9: Gràfica de la proporció de ferits Lleus, Greus i Morts al Dataframe de la Guardia Urbana.

La *il·lustració 9* mostra com és de difícil veure els ferits greus i encara més el nombre de morts (en taronja). La conseqüència d'això és que els algoritmes els hi serà més complicat detectar els ferits greus i mort, ja que en el cas que l'output fos que tots els ferits són lleus, l'algoritme obtindria una precisió del 98%. A més, aquest només podrà millorar sobre aquest 2% de dades de les classes minoritàries.

Per tant, en aquests casos, els resultats no només s'hauran de guiar només per la precisió *(Precision)* de les prediccions, sinó que s'haurà de mirar sobretot el rati de *True Positives* (també anomenat *Recall* en anglés), el *F1-Score* i les matrius de confusió que se'n derivin.

2.3.4 Tècniques per combatre les classes descompensades

Per tractar aquestes mostres descompensades, s'han emprat tècniques de *Over i Under Sampling* com SMOTE (Mordant et al., 2002) o la seva variant SMOTE-Tomek (Batista, Prati, & Monard, 2004).

També, i quan l'algoritme de Machine Learning ho permeti, s'activarà la variable interna per autoconfigurar els pesos de cada classe de forma balancejada, fet que farà que l'algoritme tingui més en consideració les classes amb menys dades.

Mirat des d'una altra perspectiva, l'algoritme és més penalitzat quan comet un error identificant les classes minoritàries que quan ho fa amb la classe majoritària després d'aplicar aquesta tècnica.

Pels altres problemes (predicció model i predicció de col·lisió) no es van haver d'usar aquesta tècnica ja que les mostres eren disperses i amb un major nombre de targets.

2.3.5 Resultats

Com a resultat de tot aquest processament de dades, els diferents dataframes dels 3 problemes d'aquest estudi han sigut els següents:

- Problema de predicció del tipus de víctimes: 20.410 entrades × 43 atributs dels anys 2016 i 2017.
- Problema de predicció del model del cotxe fugat: 18.150 entrades × 104 atributs dels anys 2016, 2017 i 2018 pels 200 models de cotxe amb més accidents.
- Problema de predicció del tipus de col·lisió dels accidents: 19.542 entrades × 28 atributs dels anys 2016 i 2017.

2.4 Anàlisi gràfica amb Tableau

En aquest apartat s'explicarà l'anàlisi amb paral·lel amb els models de Machine Learning per veure com són les dades obtingudes del procés descrit als apartats anteriors. Amb l'ajuda d'una eina de BI com és Tableau, es podrà observar quines relacions hi ha entre els atributs de les dades i veure el format d'aquestes.

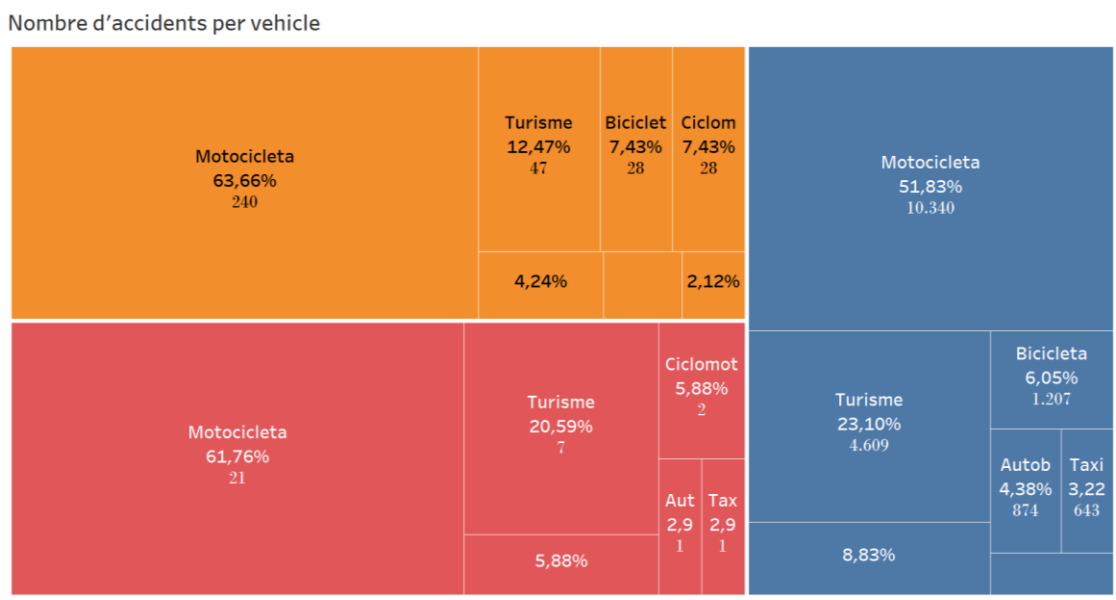
2.4.1 Gravetat dels ferits

En aquest problema es disposa d'un dataset molt dispar, cosa que fa la visualització dels resultats difícil degut a que el 98% d'aquests són ferits lleus. Per aquest motiu el que s'ha fet és normalitzar les dades per tipus de ferits.

El següents colors de la *taula 2* representen els diferents tipus de víctimes per la *il·lustració 10*:

Taula 2: Colors amb els que es relacionen els diferents tipus de ferits.

Ferit Lleu	Ferit Greu	Mort



Il·lustració 10: Treemap de les diferents victimitzacions amb els respectius vehicles amb els que es van produir.

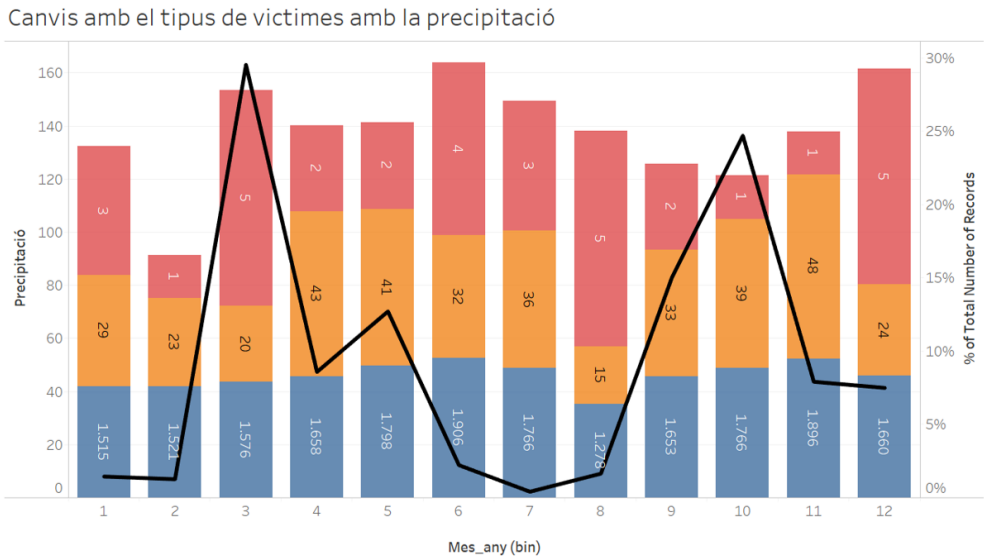
Una de les coses que es veu a primera vista és que, amb tots els tipus de ferits, el vehicle amb més accidents és la motocicleta. De fet els vehicles amb dos rodes (motocicletes, ciclomotors i bicicletes) són aproximadament el **65%** del total en accidents lleus, el **78%** en els accidents greus i el **68%** de les víctimes mortals.

Aquests són resultats semblants als de *RACC* (2018) on s'afirma que és 5 cops més probable tenir un accident greu o mortal amb moto que amb cotxe. Amb les dades *d'Open Barcelona* es pot afirmar que pel 2016 i 2017 aquesta va ser 5 cops més probable pels ferits greus i 3 pels accidents mortals.

Per contra, els turismes, que són els vehicles més utilitzat a Barcelona (aproximadament un 70% dels vehicles de Barcelona són cotxes, segons el mateix estudi del RACC), pateixen menys accidents i sempre amb percentatges més baixos de ferits que les

motocicletes. Es creu que aquest fet és degut a l'estil de conducció dels vehicles de dues rodes i sobretot a tenir menys protecció que altres vehicles amb xassís.

Un factor que es considerava rellevant a l'hora de predir el nombre d'accidents i la seva gravetat era la pluja, com afirmava l'estudi de Tamerius et al. (2016). Però la *il·lustració 11* mostra que aquesta no té un efecte rellevant sobre els accidents i el seu tipus de víctimes, com a mínim a la ciutat de Barcelona. El gràfic a continuació està normalitzat per cada any, és a dir, que la suma dels mesos per cada tipus de ferit és de 1.



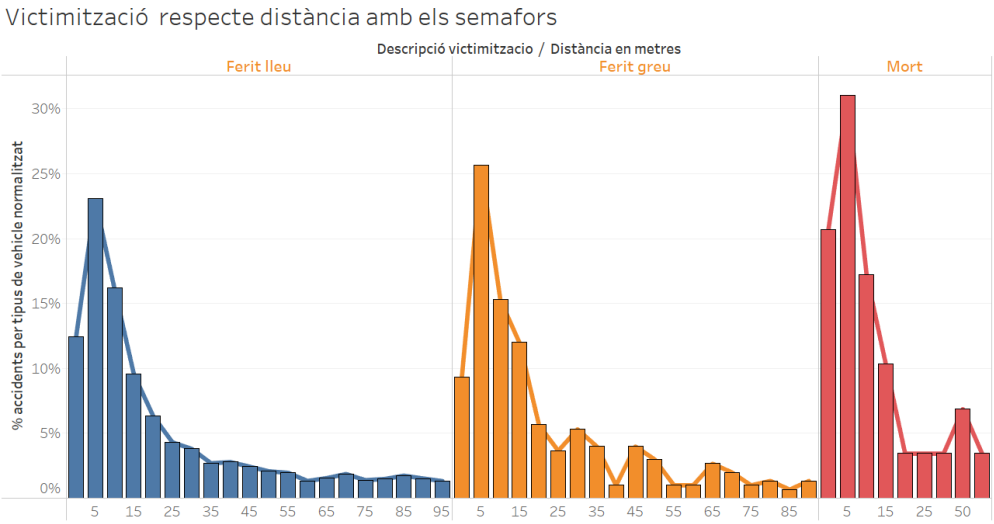
Il·lustració 11: Gràfic de barres sobreposades amb els tipus de víctimes i la precipitació per mm² mensualment.

La línia negra representa la suma de la precipitació en mm^2 per cada mes. S'observa que els mesos més plujosos com són març i octubre no tenen un increment proporcional de ferits amb la precipitació, segons les dades de 2016 i 2017.

Per exemple, el mes més plujós, març, té 5 morts però el segon mes més plujós (octubre) té només una víctima mortal. Això pot ser degut al baix nombre d'entrades de víctimes mortals ja que com s'observa, el nombre màxim de víctimes és de 5 per un mes donat.

Però amb més dades com són pel cas de ferits lleus i greus no sembla haver-hi cap relació. Per exemple, el mes de juny que és el mes amb més percentatge de víctimes, tot i no ploure gairebé gens. Per contra, el febrer, que té una pluviometria una mica menor que el mes de juny, és el que té menys víctimes. Per tot això s'afirma que a primera vista no sembla que la pluja tingui cap relació ni amb el nombre de ferits ni amb la gravetat d'aquests.

Per últim s'ha analitzat si la distància entre el semàfor més proper i l'accident és rellevant i si aquests atributs tenen alguna relació entre ells, dades resumides a la *il·lustració 12.*



Il·lustració 12: Histograma amb les diferents victimitzacions i la distància fins el semàfor amb metres.

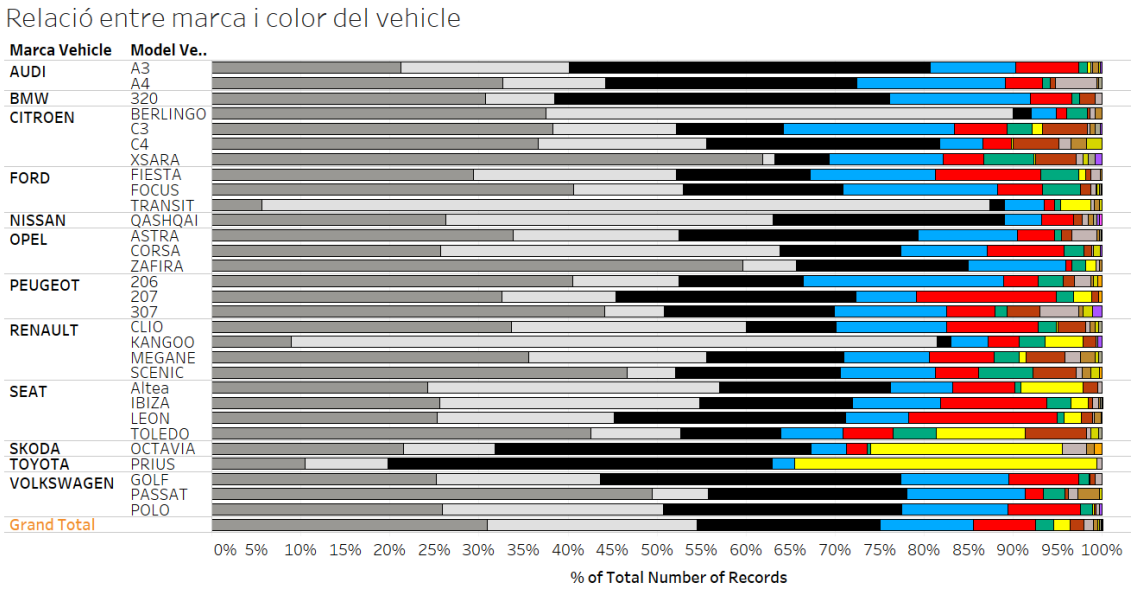
El major percentatge per cada tipus d'accident correspon a quan l'accident es dona a 5 metres de distància del semàfor. Tenint en compte que la gran part d'accidents són a l'Eixample i altres barris que segueixen el Pla Cerdà, aquest 5 metres podrien ser la zona amb poca visibilitat en la intersecció de les illes, que solen fer 40 metres de llargada.

La *il·lustració 12* ens mostra que aproximadament el **70%** de les víctimes mortals tenen lloc en els primers 10 metres. En canvi pels accidents amb ferits lleus i greus, el **50%** dels accidents passen en els 10 primers metres. És a dir, és més probable que un accident mortal passi en els 10 primers metres de distància del semàfor que en qualsevol altre rang de distàncies.

2.4.2 Models de cotxes fugats

En aquest problema la visualització va ser més difícil al tenir més variables com a target que amb el problema de les victimitzacions. En el cas dels models de cotxe fugats sobretot, serà difícil entendre totes les variables i la seva relació, degut a que hi ha més de 40 atributs en aquest dataset. L'anàlisi es centrarà, per tant, amb la relació dels models amb el color i edat que a priori semblen els atributs amb més importància per determinar el model de cotxe.

Al següent anàlisi de la *il·lustració 13* s'han seleccionat els 200 models de cotxe amb més accidents a la ciutat de Barcelona i relacionat el model d'aquests amb color. S'ha filtrat per models amb 250 accidents o més, ja que sinó es tindrien masses models de cotxe a la visualització, fet que la faria impracticable. Tot i aquest filtre, les següents representacions contenen 14.402 entrades.



Il·lustració 13: Gràfic de barres sobreposades normalitzades i amb els colors dels 30 dels 200 models de cotxe.

Al primer anàlisi s'observen les entrades normalitzades per models de les marques, on el color de la barra correspon al color del vehicle i que tots els colors sumen una probabilitat del 100% de les entrades.

A primera vista, el color més comú pels models de cotxe és el gris, amb una mitjana del **31%** dels vehicles tenint aquest color. El segueixen el blanc amb un **24%** i el negre amb un **21%**. Junts sumen un 75% del total.

D'aquest llistat de models destaquen els grocs del Toyota Prius, Skoda Octavia i Seat Toledo, tot i el reduït nombre de cotxes grocs que hi ha al dataset. Això el que ens apunta és que, amb més probabilitat, si es té una col·lisió amb un cotxe groc, serà d'algun d'aquests models. Es descarta que aquests vehicles poguessin ser taxis ja que el dataset diferencia entre cotxes privats i taxis. Però és possible que alguns dels agents hagin classificat erròniament les entrades com a cotxes i provocat aquesta nombre tan gran de sobretot Toyota Prius i Skoda Octavia de color groc.

També comentar la relació que tenen els cotxes no esportius (com familiars i furgonetes) amb els color més estàndards com el gris, blanc i negre. Un exemple és el del Nissan Qashqai o el Citroen Berlingo, que són en gran majoria grisos, blancs o negres. Denotant potser que aquells cotxes familiars o de feina no solen ser de colors llampants com sí ho és per exemple, un cotxe menys familiar com és el Seat Toledo.

A continuació s'analitza a la *il·lustració 14* la relació de les dades entre els mateixos models de cotxe i l'antiguitat de carnet que tenien les persones implicades en els accidents.

Relació entre marca i antiguitat carnet

Marca Vehicle	Model Ve..	Antiguitat carnet (bin) 0	10	20	30	40	50
AUDI	A3	177	143	131	61	24	9
	A4	129	122	118	47	50	21
BMW	320	70	47	64	37	18	7
CITROEN	BERLINGO	81	101	91	53	38	2
	C3	125	96	80	37	27	4
	C4	133	129	96	64	30	12
	XSARA	148	71	95	40	27	5
FORD	FIESTA	187	109	57	21	9	9
	FOCUS	244	213	127	72	37	9
	TRANSIT	183	172	105	76	22	4
NISSAN	QASHQAI	49	80	73	77	36	4
OPEL	ASTRA	297	185	116	52	41	17
	CORSA	214	139	59	48	23	14
	ZAFIRA	138	112	80	53	11	1
PEUGEOT	206	159	132	63	23	11	7
	207	94	71	47	21	13	12
	307	98	134	81	47	10	13
RENAULT	CLIO	208	135	88	50	25	5
	KANGOO	110	150	82	55	24	2
	MEGANE	297	180	136	103	61	18
	SCENIC	79	80	85	44	21	10
SEAT	Altea	338	256	155	107	42	5
	IBIZA	604	388	178	90	55	17
	LEON	356	353	128	53	41	9
	TOLEDO	241	136	97	64	35	3
SKODA	OCTAVIA	350	190	161	115	55	4
TOYOTA	PRIUS	346	207	173	168	66	7
VOLKSWAGEN	GOLF	467	328	234	136	51	23
	PASSAT	68	58	77	58	45	16
	POLO	311	168	105	59	16	17

Il·lustració 14: Heatmap dels 30 models més comuns normalitzat pels models de cotxe.

Els colors que es veuen a la imatge superior estan normalitzats per model novament, és a dir de forma horitzontal. El color blau marca on hi ha els majors percentatges de dades mentre que el taronja apagat mostra on n'hi ha menys entrades.

Primer de tot es veu l'abundància de blau per l'antiguitat de carnet de 0-9 anys. Això apunta que la majoria dels accidents es produeixen amb conductors amb menys anys d'experiència i que com més antiguitat de carnet menys accidents han sigut registrats, ja que tota la part de la dreta, on els conductors són més sènior, és taronja apagada.

Els cotxes amb més accidents són amb aquells models que tenen uns consumidors objectiu més joves. Aquest fet és visible amb models com per exemple el Seat Ibiza o el Volkswagen Golf i Polo, on la majoria dels accidents registrats amb aquests cotxes són de gent entre 0 i 10 anys d'experiència al volant.

Els cotxes més familiars tenen accidents amb conductors més sènior (20 o més anys de carnet). Aquest segment pertany als conductors de models més familiars com el Reault Scenic, el Nissan Qashqai o amb cotxes de marques cares i que el joves normalment no es poden permetre, com són AUDI o BMW. Per tant, es veu aquesta segmentació amb l'edat dels conductors accidentats.

Com a resultat, la probabilitat de que una persona amb 30 anys de carnet tingui un accident amb un Renault Scenic és més alta que no pas amb Renault Clio (model mes esportiu).

A continuació es pot veure el mateix gràfic a la *il·lustració 15* però els colors normalitzats per edat, així es pot distingit amb claredat els models més comuns segons l'antiguitat de carnet del conductor.

Relació entre marca i antiguitat carnet

Marca Vehicle	Model Ve..	Antiguitat carnet (bin) 0	10	20	30	40	50
AUDI	A3	177	143	131	61	24	9
	A4	129	122	118	47	50	21
BMW	320	70	47	64	37	18	7
CITROEN	BERLINGO	81	101	91	53	38	2
	C3	125	96	80	37	27	4
	C4	133	129	96	64	30	12
	XSARA	148	71	95	40	27	5
FORD	FIESTA	187	109	57	21	9	9
	FOCUS	244	213	127	72	37	9
	TRANSIT	183	172	105	76	22	4
NISSAN	QASHQAI	49	80	73	77	36	4
OPEL	ASTRA	297	185	116	52	41	17
	CORSA	214	139	59	48	23	14
	ZAFIRA	138	112	80	53	11	1
PEUGEOT	206	159	132	63	23	11	7
	207	94	71	47	21	13	12
	307	98	134	81	47	10	13
RENAULT	CLIO	208	135	88	50	25	5
	KANGOO	110	150	82	55	24	2
	MEGANE	297	180	136	103	61	18
	SCENIC	79	80	85	44	21	10
SEAT	Altea	338	256	155	107	42	5
	IBIZA	604	388	178	90	55	17
	LEON	356	353	128	53	41	9
	TOLEDO	241	136	97	64	35	3
SKODA	OCTAVIA	350	190	161	115	55	4
TOYOTA	PRIUS	346	207	173	168	66	7
VOLKSWAGEN	GOLF	467	328	234	136	51	23
	PASSAT	68	58	77	58	45	16
	POLO	311	168	105	59	16	17

Il·lustració 15: Heatmap normalitzat per l'antiguitat de carnet del conductor.

2.4.3 Tipus de col·lisions

El problema del tipus de col·lisions també és de classificació però en aquest cas hi ha menys possibles targets, un total de 9. Aquests són els tipus de col·lisió que s'han cregut més importants a l'hora de fer l'anàlisi degut a que són les més rellevants i les més nombroses com hem vist a la *il·lustració 4*.

Tipus de col.lisió per vehicle

Descripció tipus accident	Descripció tipus vehicle Autobus	Bicicleta	Ciclomotor	Furgoneta	Motocicleta	Taxi	Turisme
Abast	159	87	566	348	3.056	317	3.200
Abast multiple	14		29	258	343	174	2.698
Atropellament	119	436	195	186	1.347	205	896
Caiguda (dues rodes)	66	75	347	2	2.024	4	35
Caiguda int. vehicle	1.147	2	1	6	15	2	25
Col·lisió frontal	13	80	31	28	147	19	209
Col·lisió fronto-lateral	293	650	533	211	2.653	430	1.916
Col·lisió lateral	586	198	641	47	4.336	82	719
Xoc contra element estàtic	20	10	58	54	275	23	645
Grand Total	2.417	1.538	2.401	1.140	14.196	1.256	10.343

Il·lustració 16: Heatmap dels tipus de col·lisions amb els tipus de vehicle.

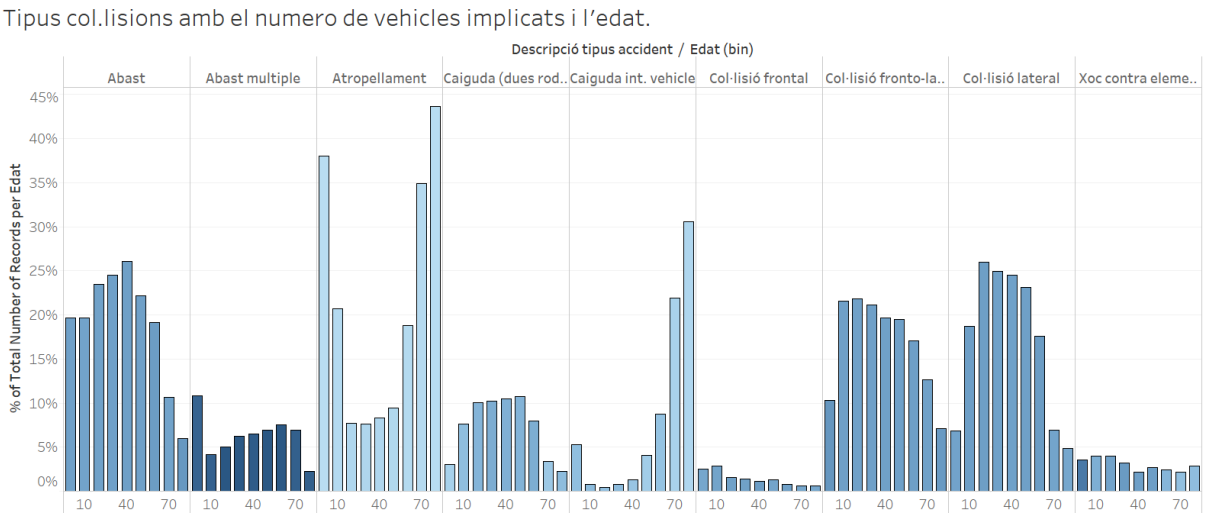
La *il·lustració 16* ha estat normalitzada per tipus de vehicle, amb el color taronja fosc on hi ha més dades i amb taronja clar on n'hi ha menys.

Per exemple, distingim que amb l'autobús l'accident més comú és la caiguda a l'interior del vehicle que compta amb casi la meitat de les dades registrades per aquest vehicle. Al següent gràfic, *il·lustració 17*, es podrà desglossar les col·lisions més freqüents per edat i veure que aquests es produeixen sobretot amb gent d'edats avançades al caure dins de l'autobús.

Un altre vehicle interessant és la bicicleta, que amb 1.538 registres, té aproximadament el **30%** d'aquests produint atropellaments, el percentatge d'atropellaments per vehicle més alt. Aquest fet es creu, amb gran certesa, es degut a que aquests vehicles no fan casi soroll al desplaçar-se i donat que a més comparteixen la vorera amb els vianants poden produir més atropellaments que el vehicles que circulen per la calçada.

Altres vehicles de dues rodes com el ciclomotor o la motocicleta tenen en comú que solen tenir accidents amb col·lisions laterals o frontó-laterals degut a canvis de carril. També poden patir col·lisions per abast quan un el vehicle de davant frena de cop.

En canvi pels cotxes la col·lisió més freqüent a la ciutat de Barcelona és l'abast. A més, no és tant comú que aquests pateixin col·lisions laterals o fronto-laterals, com sí que ho és amb els vehicles de dues rodes. El **32%** dels accidents de cotxe són produïts per abast.

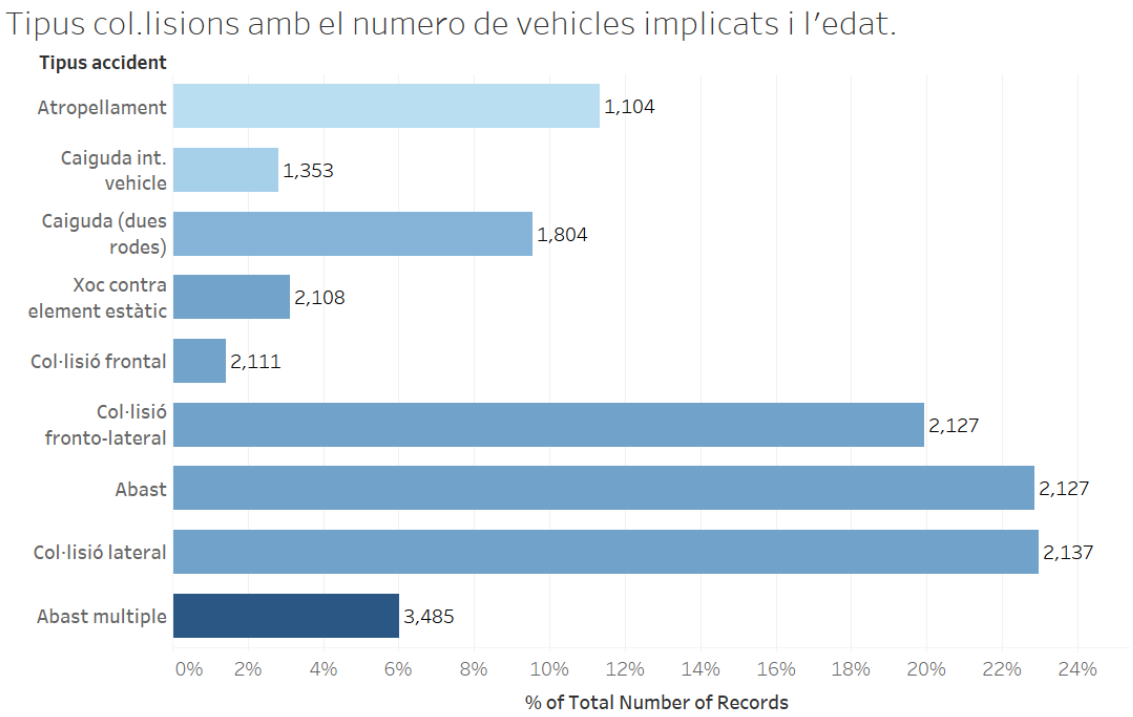


Il·lustració 17: Histogrames normalitzats per tipus de col·lisió amb la variable d'edat de l'afectat.

Al següent anàlisi (*il·lustració 17*), tenim els accidents normalitzats per cada bloc d'edat. És a dir, els percentatges de les edats de 0 a 9 anys sumen 100% amb la suma de tots els tipus de col·lisió. D'aquesta manera es veu quin és el tipus més freqüent de col·lisió per cada edat. El color fa referència a la mitjana de cotxes implicats per les diferents col·lisions.

S'aprecia que hi ha 3 tipus de tendències principalment. La primera i més sorprenent és que tant els atropellaments com les caigudes a l'interior del vehicle són més freqüents al col·lectius de **0 a 9** i de majors de **70 anys**. Es creu que pot ser degut a les menors capacitats físiques i de reduïda visió del col·lectiu d'edat avançada i degut a la seva poca alçada i moviments ràpids dels nens. A més de que aquests grup condueixen amb menys o nul·la freqüència.

En canvi, el grup d'edat entre **20 i 60 anys** experimenta una augment amb la freqüència dels tipus de col·lisió d'abast, abast múltiple, caiguda de dues rodes, col·lisió lateral i col·lisió frontó-lateral. Aquests increment és certament degut a que són aquestes les edats on més es condueix i per tant la probabilitat de patir un accident d'aquest tipus augmenta. Pel contrari, col·lectius que no condueixen o condueixen menys, tenen accidents amb autobusos o sent vianants amb més freqüència.



Il·lustració 18: Gràfica de barres amb el percentatge del total dels tipus de col·lisió, acolorit amb la mitjana de vehicles implicats a la col·lisió.

Per últim, la *il·lustració 18* serveix per veure que hi ha 3 classes d'accidents si es mira des del punt de vista del nombre de vehicles implicats, com marquen els diferents tons de blau.

Primer, els atropellaments i les caigudes a l'interior del vehicle normalment involucren **només un vehicle**. A continuació ve el grup on estan involucrats **aproximadament 2 vehicles**, com és el grup que ve des de la caiguda amb dues rodes fins a la col·lisió lateral. Finalment, l'abast múltiple és el que té més vehicles per un mateix codi d'accident, amb una mitjana de **3,5 vehicles** per col·lisió.

Per acabar, veure també amb la il·lustració anterior que les col·lisions més comunes a la ciutat de Barcelona són les col·lisions laterals, frontó-laterals i l'abast. A més, si es torna a la primera gràfica (*il·lustració 16)*, observem que les col·lisions laterals o frontó-lateral són comunes amb els vehicles de dues rodes, mentre que l'abast és més freqüent amb cotxes. I per últim destacar que el tipus de col·lisions poden estar associades a diferents col·lectius d'edat (*il·lustració 17*).

2.5 Data Modelling usant Machine Learning

Un cop feta tota la neteja de les dades, la seva preparació i anàlisi, és hora de començar a utilitzar algoritmes de Machine Learning per tal de predir els tres tipus de problemes d'aquest treball.

El que s'ha fet ha estat intentar predir cada un dels 3 resultats dels problemes, tant amb els algoritmes de Random Forest com amb Xarxes Neuronals, per tal de comparar el resultats dels dos i veure quin n'obté de millors. Aquests algoritmes van ser triats degut a que són dels millors a l'hora de solucionar problemes de classificació amb més d'un target com apunt l'estudi de Yuan et al. (2017).

Tots els Datasets dels problemes ha estat separats en un 0.75 a Training i un 0.25 a Testing. Es va considerar que l'ús d'un set de validació no seria adequat ja que, primer, són problemes diferents i segon, perquè la finalitat d'aquest experiments és veure quin dels dos algoritmes prediu millor el mateix problema.

Degut a la capacitat de l'ordinador utilitzat, l'algoritme de Xarxes Neuronals va ser computat a la plataforma núvol de Google utilitzant codi en llenguatge Python amb la llibreria de *Keras* a Google Colaboratory. En canvi els algoritmes de Random Forest van ser executats en local amb Jupyter Notebook amb la llibreria de *Sklearn*. En aquest estudi es mostren només les millors configuracions dels algoritmes Random Forest i Xarxes Neuronals obtingudes.

Finalment a l'apartat de discussió, farem una comparació dels dos models analitzats per veure quin algoritme és millor per cada tipus de problema.

2.5.1 Predicció de la gravetat dels ferits

Per aquest problema s'usa un classificador base *(Dummy Classifier)* que segueix l'estratègia de predir sempre la classe més comuna, l'estratègia prior. Aquest serà amb el que comprarem els resultats dels nostres algoritmes.

L'Accuracy del Dummy Classifier per tant és el mateix que el percentatge de la classe més comuna, un 0.98060.

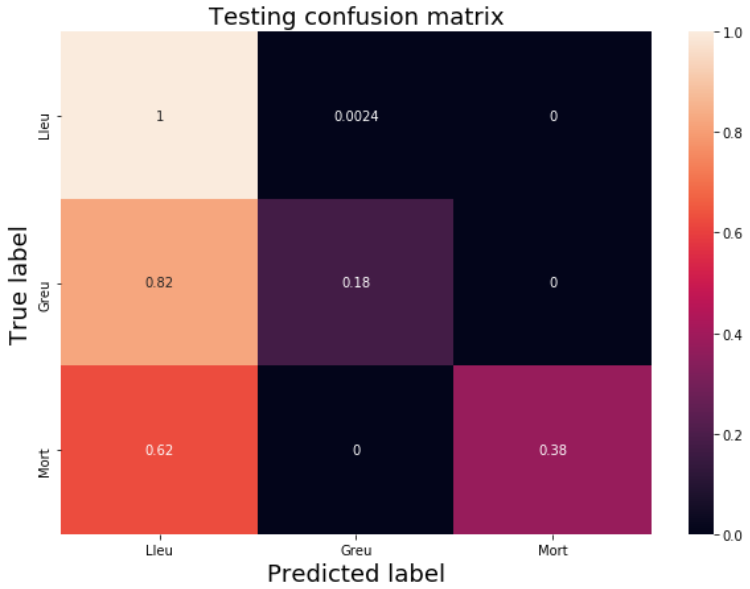
2.5.1.1 Random Forest

Al ser aquest un problema amb classes descompensades es va aplicar una tècnica *d'Over i Undersampling* coneguda com *SMOTE-Tomek* de manera que les tres classes comptessin amb el mateix nombre d'entrades cada una i així fer que totes tinguessin el mateix pes per l'algoritme de Random Forest.

També es va aplicar la tècnica de la graella de busca *(Grid Search)* amb diferents paràmetres com el nombre d'estimacions ([200, 500, 1000, 2000]) i la profunditat màxima de les branques de l'algoritme ([4, 7, 10, 30]). Tot això va anar acompanyat d'una *5-Fold CrossValidation.*

Per últim, el paràmetre del pes de les classes *(class weights)* de l'algoritme ha sigut posat a balancejat *(balanced*) per combatre encara més el problema de *Class Unbalance* que té aquest target.

Un cop executat l'algoritme, el resultat de la matriu de confusió aconseguida per l'algoritme sobre el set de test ha estat el de la *il·lustració 19*.



Il·lustració 19: Matriu de confusió del Random Forest pel tipus de victimització.

S'observa que el classificador és perfecte pel que fa a la precisió dels True Possitives pels accidents lleus, però també classifica i confon entrades d'altres gravetats com a ferits lleus, degut a que aquesta és la classe més comuna. És a dir que tot i els tractaments de dades, l'algoritme encara continua classificant la majoria d'instàncies com a ferit lleu al *testing set*.

Però també gràcies a tot el processament de les dades s'ha aconseguit que la diagonal secundària de la matriu no sigui tota formada per zeros, sobretot per les dues classes minoritàries. Els accidents greus s'han classificat correctament en el **18%** de les ocasions i els morts en el **38%.** La resta han sigut classificats incorrectament com a ferits lleus com hem explicat anteriorment.

Per problemes com aquests és necessari mirar altres mètriques que no siguin només l'Accuracy del model com afirma Brownlee (2018). Per aquest motiu, altres indicadors com els de la *Taula 3* sobre el *Testing set* han sigut usats:

Taula 3: Puntuació de Precision, Recall i F1-Score sobre l'algoritme de Random Forest pels diferents ferits.

	Precision	Recall	F1-Score	Support
Lleus	0.98	1.00	0.99	5004
Greus	0.64	0.18	0.28	91
Víctimes mortals	1.00	0.38	0.55	8
Weighted Avg.	**0.98**	**0.98**	**0.98**	**5103**

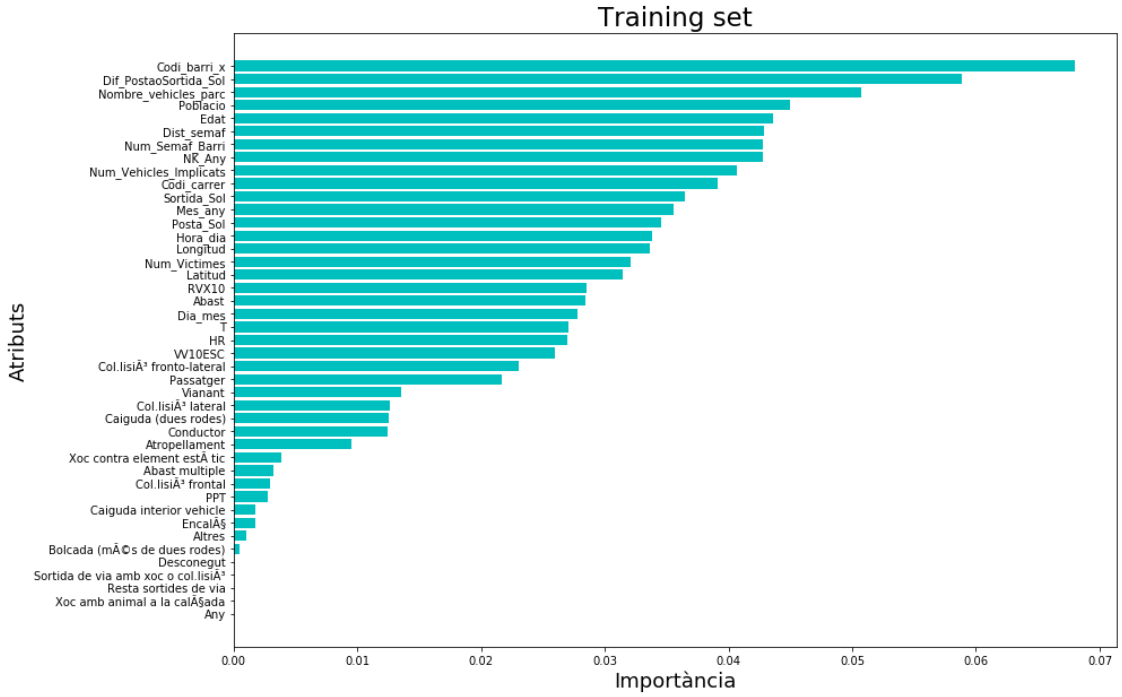
S’observa que els tres tipus de targets no tenen el mateix nombre d’entrades cada un com es pot veure a l’apartat de *Support*. Això és degut a que només s’ha usat la tècnica de *SMOTE-Tomek* per entrenar el classificador, és a dir, només al set d’entrenament per així no alterar la mostra del set de test i tenir un resultat no esbiaixat.

El classificador ha obtingut una bona *Precision* però sobretot amb els ferits greus i els morts, el *Recall* per altra banda ha estat dolent. Això ha fet disminuir el seu *F1-Score*, (la mitjana harmònica de la *Precision* i el *Recall*) significativament.

Els millors resultats han estat els de la classificació dels ferits lleus amb un *F1-Score* del 0.99 i que degut a que són el 98% de les entrades, tenen un gran pes a l’hora de computar la mitjana ponderada.

Finament el classificador ha obtingut una *Accuracy* del **0.99996** i del **0.98265** sobre el se t d’entrenament i el set de test, respectivament.

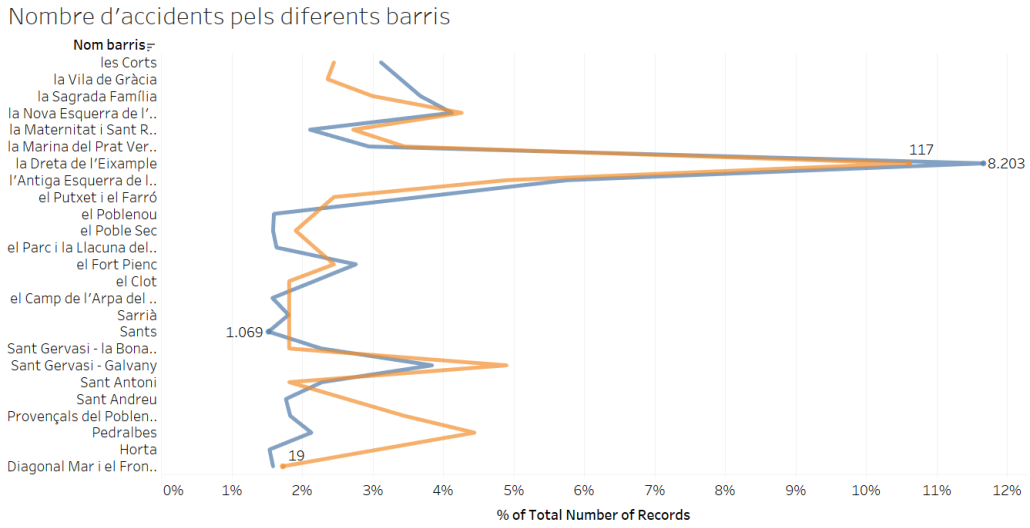
Per últim, s’analitza la importància relativa dels atributs segons el nostre model de *Random Forest*. Aquesta és un llistat dels atributs i com d’importants han estat a l’hora de fer la separació de les branques de l’algoritme, funció exclusiva dels algoritmes en arbre que es mostra a la *il·lustració 20*.



Il·lustració 20: Gràfica de barres sobre la importància dels atributs segons l’algoritme.

Aquesta funció permet observar que el Random Forest considera el codi del barri important a l’hora de decidir la gravetat del ferit, atribut que no havia sigut considerat a l ’apartat d’anàlisi.

Aquest fet pot ser atribuït a que sobretot al codi de Barri número 7 apareix en molts dels accidents. Aquest codi és el del barri de la nova dreta de l’Eixample, que com s’ha vist a l’apartat *2.1 Data Understanding*, és el districte (barris Eixample) amb més accidents de tota la ciutat.



Il·lustració 21: Percentatge sobre el total d'accidents (Lleus i Greus) pels barris amb més accidents. No es mostren els accidents amb víctimes mortals degut als escassos registres.

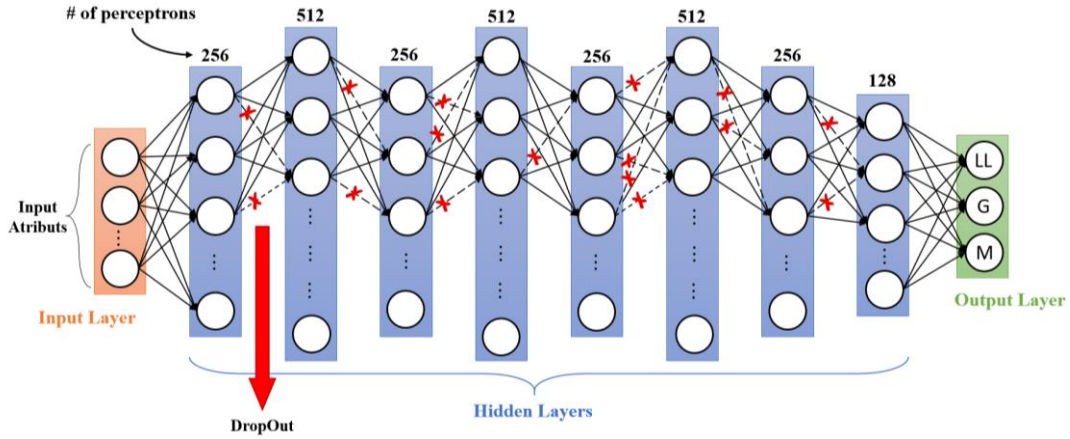
Una posterior anàlisi més detallada amb Tableau resulta amb la *il·lustració 21*, on el barri amb més accidents de llarg és el de la Dreta de l'Eixample, tant per ferits lleus (blau), com pels ferits greus (taronja). Només els 6 barris de, la nova i la vella esquerra de l'Eixample, la dreta de l'Eixample, Sant Gervasi - Galvany, Pedralbes i les Corts sumen el **30,3%** dels accidents lleus i el **34,2%** dels accidents greus.

Tenint en compte que Barcelona té un total de 73 barris, els anterior són només el **8,2%** dels barris i ocupen el **10,24%** de la superfície de la ciutat segons dades de l'ajuntament (1.046,9 hectàrees) però és on tenen lloc mes del **30%** dels accidents.

A la importància de l'atribut del barri, segons la *il·lustració 20*, el segueixen altres com la diferencia amb hores entre l'accident i la sortida o posta de sol, el nombre de vehicles en aquell barri, la població del mateix, l'edat dels implicats i la distància entre l'accident i el semàfor, per anomenar alguns dels atributs més importants.

2.5.1.2 Xarxes Neuronals

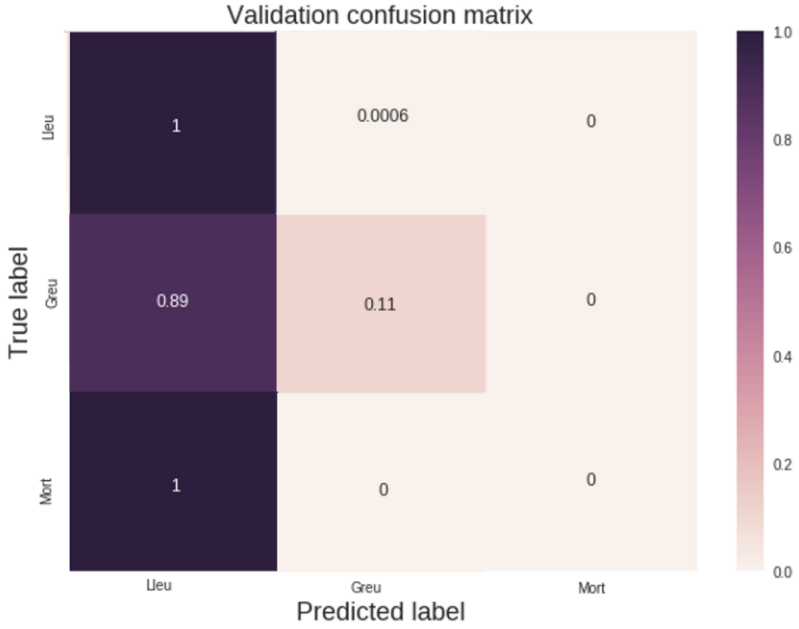
En aquest tipus d'algoritme s'ha utilitzat un model seqüencial amb la funció d'activació *ReLu* a les capes intermèdies i *Softmax* a la capa de sortida *(Output Layer)*. Altres paràmetres usats són 0.35 de *Drop-out* per evitar *Overfitting* i 8 capes intermèdies *(Hidden Layers)* amb normalització en batch a cada una d'aquestes capes amb la configuració que es mostra a la il·lustració 22.



Il·lustració 22: Esquema de la Xarxa Neuronal pel problema de predicció de ferits.

A més, s'ha utilitzar un punter (*checkpointer)* per guardar la configuració amb els millors pesos *(weights)* per obtenir la millor precisió possible. La Xarxa Neuronal va ser compilada amb l'optimitzador *Adam (o Adaptive Momentum Estimation)* amb els paràmetres per defecte.

Amb 50 *Epochs* (50 front i backpropagations) s'aconsegueix el resultat següent:



Il·lustració 22: Matriu de confusió per l'algoritme de Xarxes Neuronals

En aquest cas, la Xarxa Neuronal classifica gairebé totes les entrades com a ferit lleu i és només amb els ferits greus que en classifica el **0.11** correctament. No veiem la segona diagonal de *True Positives* en aquest cas ja que la xarxa neuronal ho classifica tot com la classe més comuna i poques instàncies de greus són classificades correctament. A més l'algoritme no és capaç d'encertar cap de les víctimes mortals al ser aquestes instàncies minoritàries amb tan poc pes al set de test.

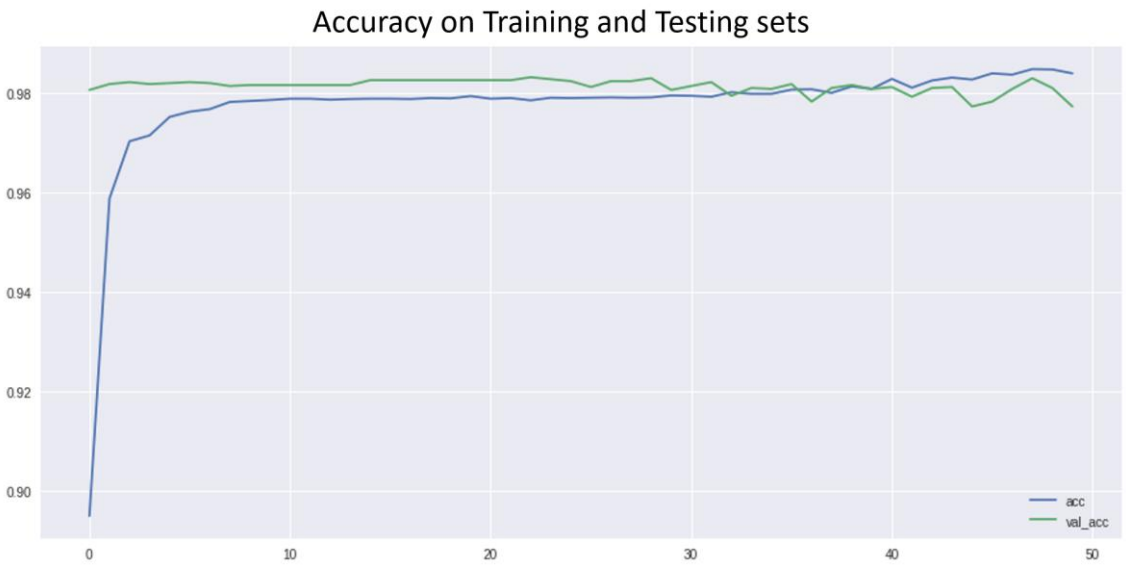
Taula 4: Puntuació de Precision, Recall i F1-Score sobre l'algoritme de Xarxes Neuronals pels diferents ferits.

	Precision	Recall	F1-Score	Support
Lleus	0.98	1.00	0.99	5009
Greus	0.77	0.11	0.19	90
Víctimes mortals	0.00	0.00	0.00	4
Weighted Avg.	**0.98**	**0.98**	**0.98**	**5103**

A la *Taula 4* veiem com l'algoritme aconsegueix una excel·lent puntuació de 0.99 pel F1-Score, però pels altres targets aquesta és casi nul·la. Destaca sobretot les víctimes mortals que tenen una *Precision* i un *Recall* de 0 al no haver-se classifica cap instància com a *True Positive*.

Tot i això, obtenim concretament una *Accuracy* del **0.986803** al *training set* i del **0.982951** sobre el *testing set*.

A continuació s'analitzarà des del punt de vist de *l'Accuracy* a mesura que es produeixen les *Epochs*, com es descriu a la *il·lustració 24*.



Il·lustració 24: Evolució de l'Accuracy dels Testing i Training sets respecte les epochs.

La precisió sobre el set d'entrenament (Blau) supera la del set de test (Verd) però sense arribar a nivells *d'Overfitting*. *L'Accuracy* del set de test no es mou quasi res, degut a que té 5.000 mostres de ferits lleus i només 90 entrades de ferits greus i 4 de morts. Això fa que el classificador obtingui millors resultats focalitzant-se amb els ferits lleus, degut a que són la gran majoria de l'espectre, cosa que s'ha vist a la matriu de confusió.

2.5.1.3 Discussió

La primera conclusió que s'ha arribat ha estat que és molt complicat lluitar contra un problema de *Class Unbalance* tan sever. La segona és que es necessitarien més dades sobre víctimes mortals per tenir una anàlisi més robust a les variacions, ja que (afortunadament) només el 0,1% de tot el dataset eren víctimes mortals.

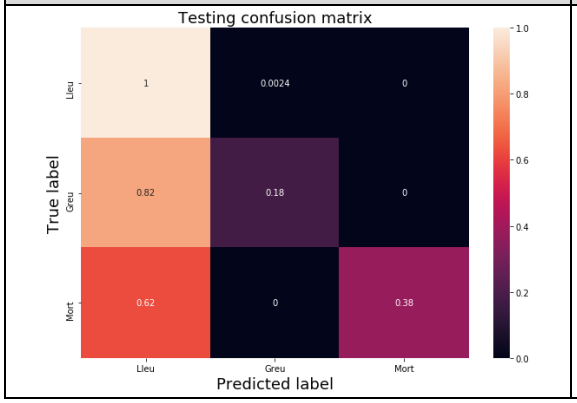
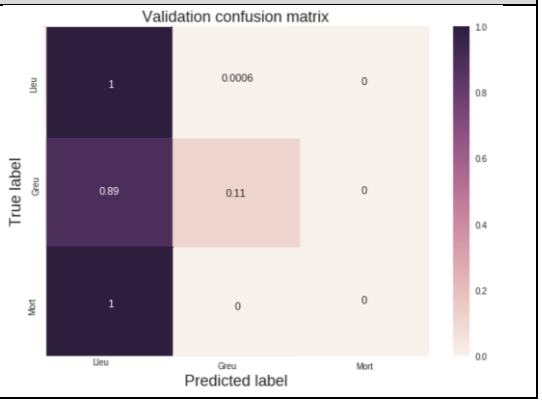
A la *Taula 5* s'analitzen els resultats dels dos classificadors respecte la *Accuracy*:

Taula 5: Resum de l'Accuracy sobre els diferents algoritmes i el classificador base.

	Dummy Classifier		Random Forest		Xarxes Neuronals	
Accuracy	**Δ -**	**0.98060**	**Δ0,20%**	**0.98256**	**Δ0,24%**	**0.982951**

Amb aquests resultats, es pot afirmar que el millor classificador respecte *l'Accuracy* per aquest problema ha estat la Xarxa Neuronal Profunda obtenint el millor resultat amb un **0.982951** d'instàncies correctament classificades pel *Testing set*, encara que per un marge molt reduït. Pot semblar poca diferència entre els tres classificadors, però s'ha de tenir en ment que el 98% de les instàncies eren d'una sola classe i que és en aquest **2%** on realment han marcat la diferència els classificadors més complexos.

Taula 6: Comparativa de les matrius de confusió dels diferents algoritmes.

Random Forest	Xarxes Neuronals
	

Pel que fa a les matrius de confusió, es mostra una taula comparativa d'aquestes a la *Taula 6.* En aquesta, el Random Forest aconseguia predir correctament el **38%** de les entrades classificades com a mort i el **18%** de ferits greus.

Aquest resultat més bo que el de les Xarxes Neuronals, que han predit la gran majoria d'entrades com a ferit lleu i classificat correctament només un **11%** de les instàncies de f erits greus i cap de les de morts.

Aquest fet es pot veure clarament si mirem la comparació els *F1-Scores* d'un i altre algoritme, tal com es pot veure a la Taula 7.

Taula 7: Comparativa del F1-Score pel set de test dels algoritmes, centrant-se amb les classes minoritàries.

F1-Score	Random Forest	Xarxes Neuronals
Greus	0.26	0.19
Víctimes Mortals	0.55	0.00

Al ser aquest un **dataset amb classes descompensades aquestes mètriques tenen més pes** que en un dataset balancejat i per tant es pot afirmar que encara que la Xarxa Neuronal tingui una millor *Accuracy* (per poc), el F1-Score del Random Forest és molt millor i decanta la balança per aquest últim algoritme.

Per tant, la classificació dels ferits greus i víctimes mortals segons el *F1-Score*, l'algoritme de Random Forest és bastant superior a les Xarxes Neuronals. Pel que fa als ferits lleus, els dos *F1-Scores* són idèntics (0.98) i per tant no s'han considerat rellevants per posar-se en aquest anàlisi.

Com a conseqüència, es conclou que l'algoritme de Random Forest és més bo a l'hora de classificar correctament les diferents entrades segons s'ha vist amb el *F1-Score*, tot i que les Xarxes Neuronals ha obtingut una *Accuracy* una mica millor.

És per aquest motiu que no ens pot guiar l'anàlisi només amb la *Accuracy* de l'algoritme i s'ha de mirar altres mètriques com les matrius de confusió i *F1-Scores* per veure que no és el millor algoritme el que té una millor classificació, sobretot en problemes de classes no balancejades. I per tant, també interessa que aquests sigui capaç de predir les entrades de les classes minoritàries com sí ho fa el Random Forest.

2.5.2 Predicció del models de cotxes fugats

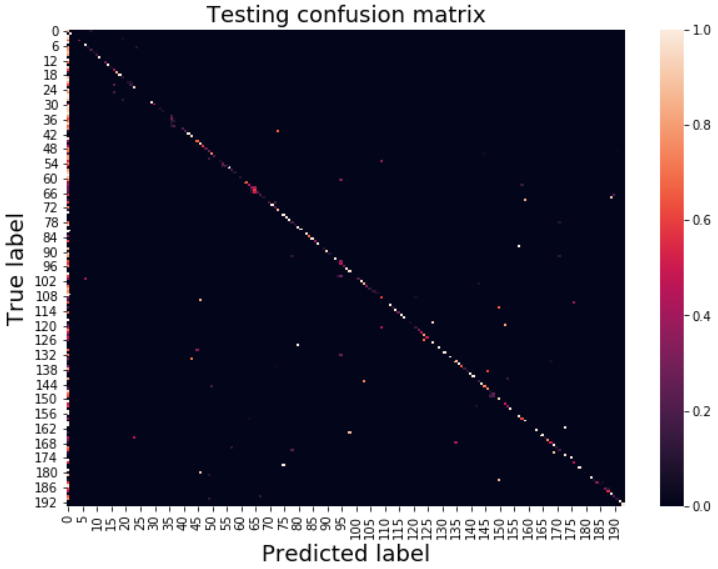
El classificador base per aquest problema ha estat fet amb un simple algoritme que, tenint en compte la marca del vehicle, responia amb el model més comú per aquella marca de cotxe. Fent ús d'aquest algoritme, *l'Accuracy* del classificador base ha estat del **0.3610**.

En aquest cas no ha sigut necessari emprar cap tipus d'eina de *Oversampling* degut a que els targets d'aquest problema eren molt més nombrosos i dispersos. Concretament per aquest problema hi ha 200 targets, que són els 200 models de cotxe amb més accidents a la ciutat de Barcelona.

2.5.2.1 Random Forest

Comencem també aquest cop fent servir un Parameter Grid per arribar a aconseguir la millor combinació de paràmetres que creïn el classificador més acurat possible. En aquest cas també hem posat diferents valors pel nombre d'estimacions ([50, 150]) i el nombre màxim de profunditat de les branques l'algoritme ([10, 30, 50]). Degut a la complexitat d'aquest problema i el més de 100 d'atributs, no s'ha pogut usar un gran nombre d'estimacions. S'ha usat també un *3-Fold Crossvalidation.*

Finalment, veiem el resultat de la matriu de confusió del classificador a la *il·lustració 25*.



Il·lustració 25: Matriu de confusió del Random Forest respecte els models de cotxe.

Podem observar clarament la diagonal secundària a la matriu de confusió, i per tant podem assegurar que el classificador té, a priori, una bona *Accuracy* respecte el gran ventall de target que té i obté un bon resultat de *True Possitive*.

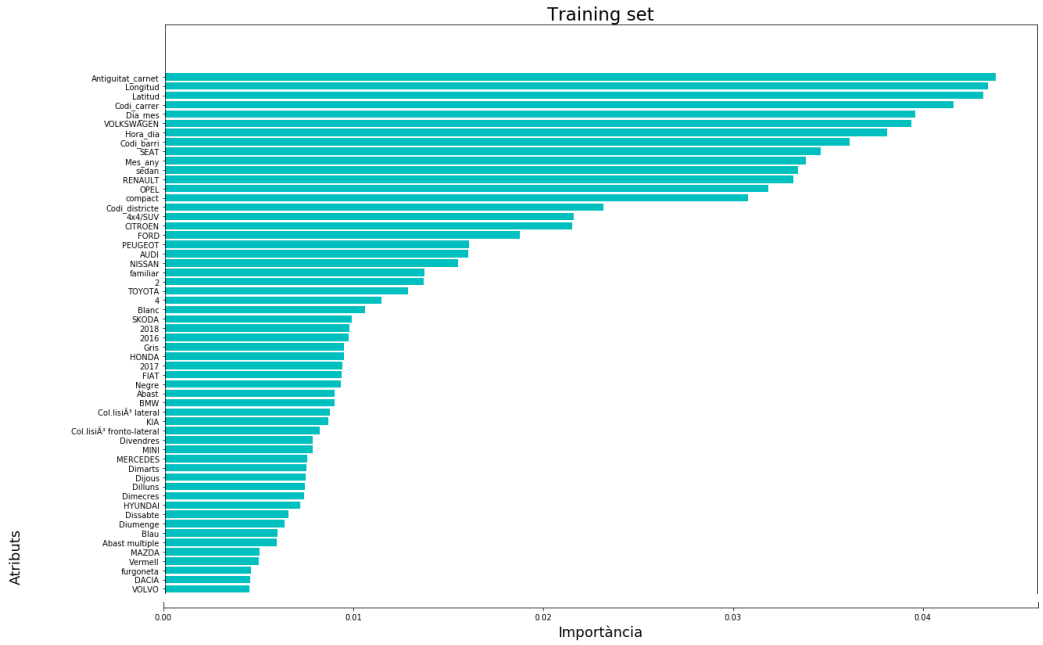
En aquest cas l'informe de classificació no el podem arribar a mostrar ja que té 200 files, una per cada un dels models de cotxe, però sí es mostrarà la mitjana ponderada d'aquest a la *Taula 8*.

Taula 8: Mitjana ponderades dels indicadors de Precision, Recall i F1-Score amb l'algoritme de Random Forest.

	Precision	Recall	F1-Score	Support
Weighted Avg.	**0.70**	**0.50**	**0.52**	**4538**

El resultat del *F1-Score* ha estat de 0.52 com a mitjana de tots els diferents models. Aquest resultat es creu és degut al gran nombre de possibles targets, que segurament han fet disminuir el percentatge de positius classificats correctament. Això ha fet disminuir el *Recall* del model, però tot i així s'obté un bon resultat pel classificador.

A continuació s'analitza la importància dels atributs respecte l'algoritme amb la *il·lustració 26.*



Il·lustració 26: Importància dels atributs segons l'algoritme de Random Forest pel problema dels models de cotxe.

S'ha de tenir en compte que la il·lustració anterior ha hagut de ser truncada ja que la imatge pels 100 atributs i la seva importància ocupaven casi tota la pàgina.

Els valors que més sovint s'han utilitzat per fer les branques han sigut atributs com l'antiguitat del carnet, que va relacionat amb l'edat de la persona, la ubicació de l'accident amb coordenades, el codi del carrer, hora del dia o algunes de les marques amb més accidents com són SEAT i Volskwagen.

Finalment obtenim una Accuracy del **0.99985** al set d'entrenament i del **0.49846** al set de test.

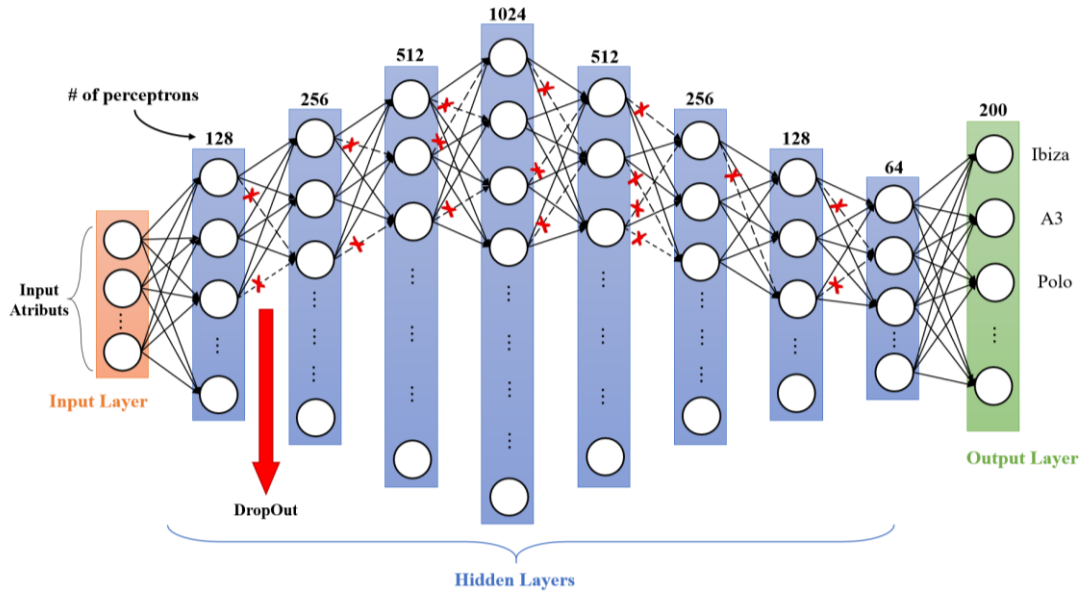
2.5.2.2 Xarxes Neuronals

S'ha emprat un model seqüencial profund amb moltes capes per així arribar a aconseguir que la xarxa sigui capaç de detectar tendències amagades a les dades de gran complexitat.

Per aquest model s'ha emprat una funció d'activació de *Tanh* a les capes intermèdies *(Hidden Layers)* que ha resultat més efectiva en aquest problema que la *ReLu*. Pel que fa al nombre de *Hidden Layers* en aquest cas han sigut de 8 també però amb més perceptrons

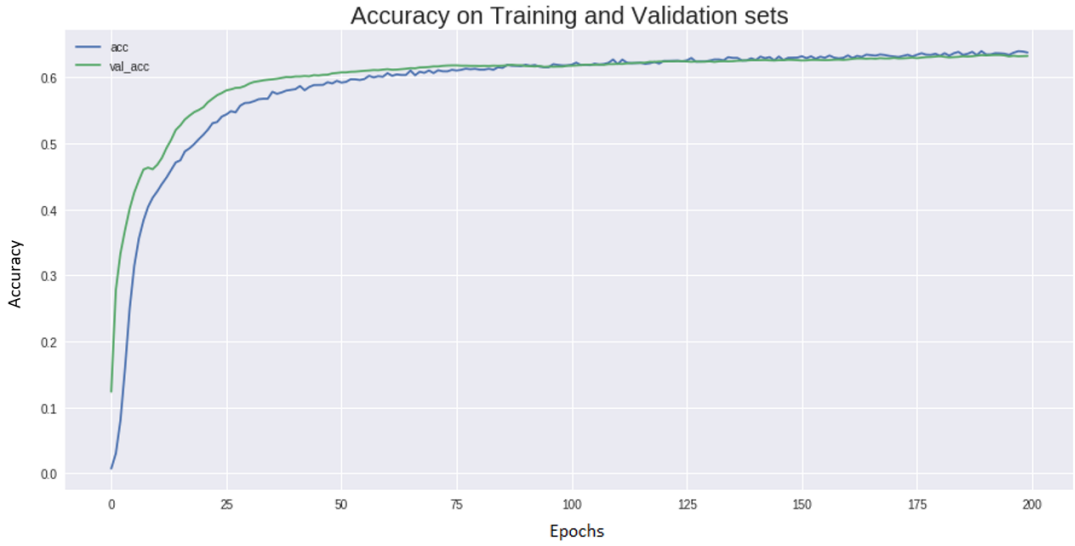
en total. S'ha usat també un *Dropout* del 0.2 un cop més per lluitar contra *l'Overfitting* i *Batch Normalization* per normalitzar cada cop totes les capes intermèdies.

Pel que fa a la funció d'activació a la capa d'output ha estat usat un *Softmax* amb un optimitzador SGD *(Stochastic Gradient Descent)* amb un ritme d'aprenentatge *(Learning Rate)* de 0.0125, un *Decay* de $1x10^{-6}$ i *Momentum Nesterov* de 0.95. L'esquema del disseny de la xarxa es mostra a la *il·lustració 27*.



Il·lustració 27: Esquema de la Xarxa Neuronal pel problema de predicció de models de cotxe.

El model va ser executat amb 200 *Epochs* i el resultat va ser el de la *il·lustració 28.*



Il·lustració 3: Evolució de l'Accuracy de l'algoritme de Xarxes Neuronals sobre el Training i Testing sets.

Aquest resultat ens mostra clarament que a mesura que el model s'entrena, aquest va guanyant més i més *Accuracy* tant amb el set d'entrenament com amb el de test. Fins al punt que es passa d'una *Accuracy* de **0.0069** al *Training set* i del **0.1235** al *Testing set*, fins convergir els dos test a els **0.64942** de *Training* i a **0.6324** del *Testing* set.

Aquest resultat, a més de ser considerat molt bo, no presenta *Overfitting* ja que tant el set d'entrenament com el set de test tenen una *Accuracy* semblant. Aquest resultat permet una predicció un **43%** millor que la *Best informed guess* del classificador base explicat anteriorment.

Degut a la similitud amb l'anterior algoritme i el no poder-se apreciar el detall els valors de a la matriu de confusió, aquesta ha sigut descartada per aquest anàlisi i també per la posterior discussió. Per últim amb l'informe de classificació es pot veure a la *Taula 9*.

Taula 9: Mitjana ponderades dels indicadors de Precision, Recall i F1-Score amb l'algoritme de Xarxes Neuronals.

	Precision	Recall	F1-Score	Support
Weighted Avg.	**0.46**	**0.55**	**0.48**	**4543**

En aquest cas, el model ha tingut una Precision no tan bona com la del Random Forest. Però tot i això, la Xarxa Neuronal obté un F1-Score prou correcte tenint en compte el gran nombre de targets de la mostra.

2.5.2.3 Discussió

S'ha obtingut un sorprenent resultat a *l'Accuracy* de la Xarxa Neuronal, que ha superat de llarg a el classificador base i a l'algoritme de Random Forest. Veiem les *Accuracies* de cada un dels models.

Taula 10: Resum de l'Accuracy i el F1-Score sobre els dos algoritmes analitzats i classificador base.

	Dummy Classifier		Random Forest		Xarxes Neuronals	
Accuracy	**Δ -**	**0.3610**	**Δ 27,57%**	**0.49846**	**Δ 42,91%**	**0.63240**
F1-Score Avg.		**-**	**Δ 7,70%**	**0.52000**	**Δ -**	**0.48000**

Si tenim en compte els resultats del problema anterior de classificació dels ferits veiem que aquest cop no estem parlant d'una millora del 0,24% sinó del **42,9%** amb les Xarxes Neuronals respecte el *Dummy Classifier*. A més obté una *Accuracy* **21,18%** millor que el model de Random Forest. Aquest fet pot ser degut a diversos factors com el fet de no tenir *Class Unbalance*, el gran nombre de targets que hi havia i el gran numero de registres des del 2016 al 2018 que han permès a les Xarxes Neuronals obtenir un patró a les dades per arribar a tenir una Accuracy del **0.6324**.

Sobre els resultats del *F1-Score* el resultat és una mica millor pel Random Forest, exactament un **7,70%**. La diferència no és significativa ja que a més no estem davant d'un dataset amb classes descompensades.

Pel que fa les matrius de confusió, al haver-hi tants targets no podem saber amb precisió on s'han confós els algoritmes, per tant, aquestes matrius només ens han resultat útils en aquest problema per comprovar que efectivament els algoritmes dibuixaven una diagonal segona, senyal que estaven classificant correctament la majoria de les instàncies com a *True Possitives*.

Com a conclusió podem afirmar que per aquest problema les Xarxes Neuronals han estat l'algoritme més efectiu per resoldre'l amb una *Accuracy* molt bona en un problema amb 200 targets i més de 100 atributs.

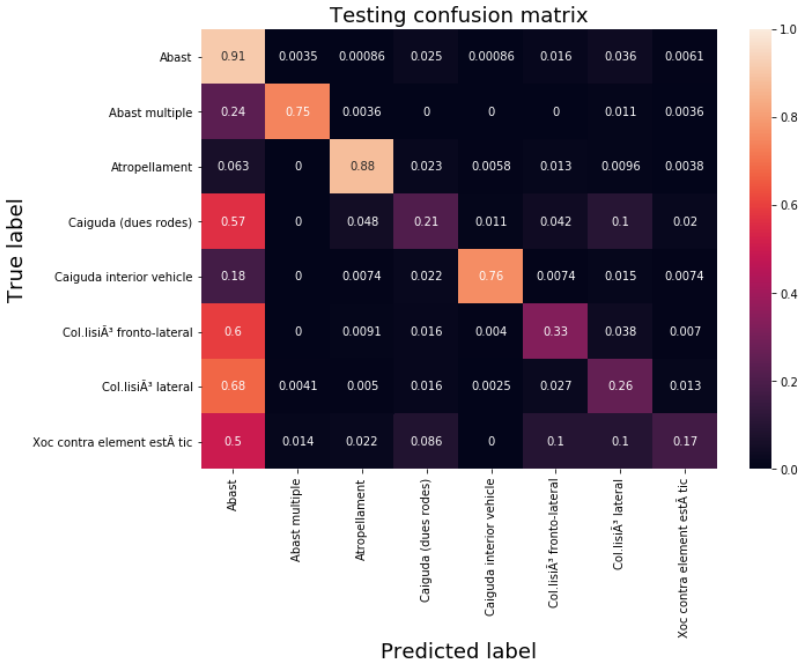
2.5.3 Predicció del tipus de col·lisió

Per a aquest problema el classificador base usat va ser el prior, és a dir, sempre respondre amb al classe més comuna del dataset. Usant aquesta estratègia s'aconsegueix una *Accuracy* del **0.2330** classificant totes les entrades com a col·lisió lateral, la classe més comuna.

2.5.3.1 Random Forest

Per aquest problema es desenvolupa un plantejament semblant als anteriors però amb un *Parameter Grid* una mica més ampli. Els paràmetres d'aquest han estat el nombre d'estimacions ([100, 500, 2.000]) i la màxima fondària de les branques ([10, 50, 200]). Tots això amb una *3-Fold Crossvalidation* amb el criteri de *Gini*.

Al executar l'algoritme, aquest genera la matriu de confusió de la *il·lustració 29*.



Il·lustració 29: Matriu de confusió de l'algoritme Random Forest pel problema de predicció de les col·lisions.

La matriu de confusió mostra la incorrecta classificació de molts dels targets que són classificats com abast, com es veu clarament a la primera columna de la matriu. Exemple d'això és el xoc amb element elàstic que es prediu correctament el 17% de les vegades, mentre el 50% de les instàncies es confon com a abast.

Pels altres targets amb un gran nombre d'instàncies com són la col·lisió frontó-lateral i la lateral, les confon també amb col·lisions d'abast. Fet que farà disminuir *l'Accuracy* del model.

Pel que fa el *F1-Score* obtenim el resultat que es pot veure a la *Taula 11*.

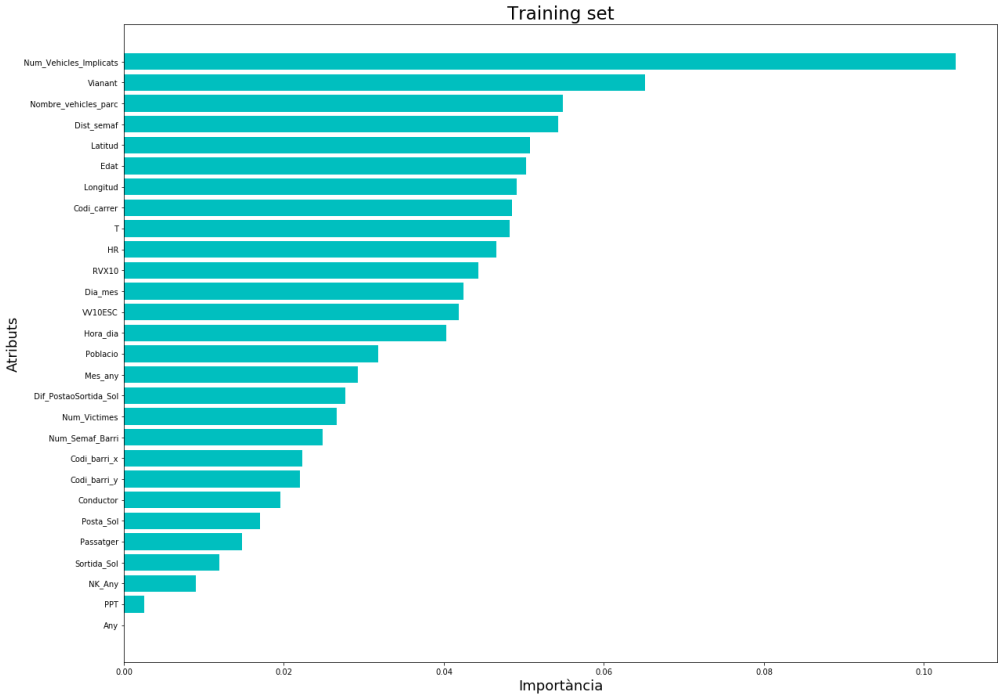
Taula 11: Mètriques de Precision, Recall i F1-Score pels diferents tipus de col·lisió per l'algoritme de Random Forest.

	Precision	Recall	F1-Score	Support
Abast	0.78	0.35	0.48	1157
Abast múltiple	0.95	0.75	0.84	279
Atropellament	0.91	0.88	0.90	520
Caiguda 2 rodes	0.51	0.21	0.30	456
Caiguda Int. Veh.	0.87	0.76	0.81	136
Fronto-lateral	0.78	0.33	0.46	993
Lateral	0.67	0.26	0.37	1206
Element estàtic	0.36	0.17	0.23	139
Weighted Avg.	**0.74**	**0.39**	**0.50**	**4886**

Aquest resultat mostra que l'algoritme té una bona *Precision* però on comet més errors és al *Recall*, té un baix rati de positiu classificats correctament. Fet que es pot veure clarament mirant la matriu de confusió de la passada il·lustració 29, on moltes de les entrades es classifiquen com a Abast encara que no ho siguin.

Sobretot tenim puntuacions de F1-Score molt baixes l'Abast, ja que com hem dit hi ha moltes instàncies que són *False Possitives.* Altres targets com són les col·lisions laterals i amb elements estàtics pràcticament no tenen *True Possitives*.

Passem ara a analitzat la importància dels atributs pel Random Forest a l'hora de fer les prediccions com es veu a la *il·lustració 30*.



Il·lustració 30: Importància dels atributs segons l'algoritme de Random Forest.

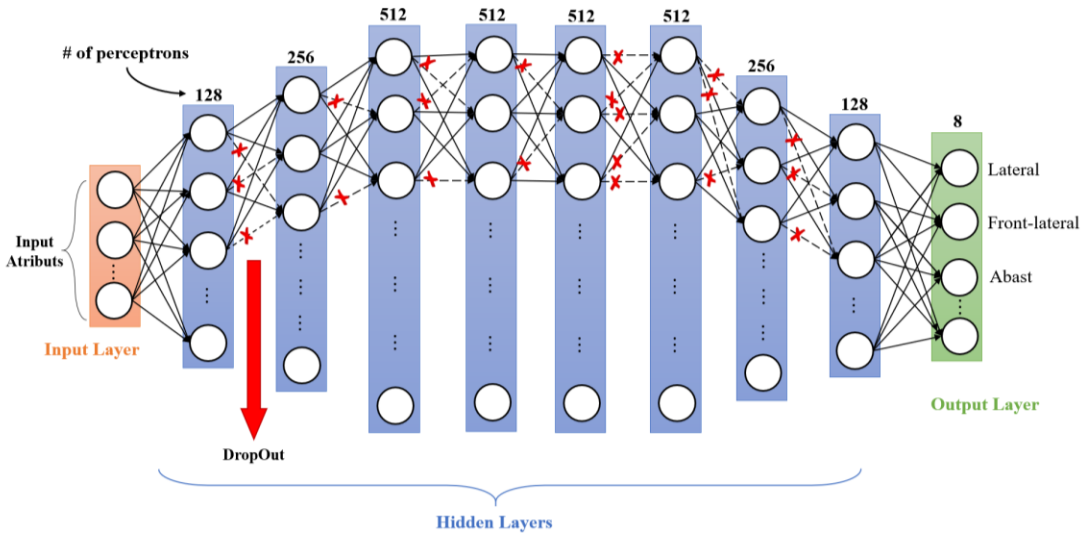
Segons l'algoritme, l'atribut més important és el nombre de vehicles, que com s'ha vist a l'apartat d'anàlisi amb Tableau, agrupava els diferent accidents en 3 blocs diferenciats

(il·lustració 18). A una distància considerable venen els vianants, col·lectiu que només pot ser accidentat si és atropellament o caiguda a l'interior del vehicle.

Finalment la Accuracy del algoritme ha estat del **0.93811** sobre el set d'entrenament i **0.39460** sobre el set de test.

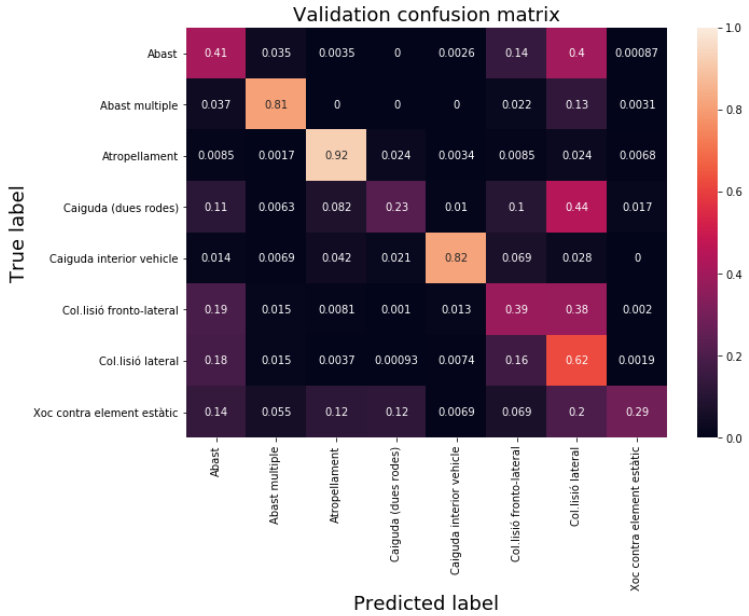
2.5.3.2 Xarxes Neuronals

Per aquest model seqüencial s'ha utilitzat la funció d'activació *Tanh* a les capes intermèdies i *Softmax* per la capa d'output. A cada capa intermèdia s'ha efectuat una *Batch Normalization* i un *Dropout* del 0.35 per evitar *l'Overfitting* del model a la mesura del possible. El model ha estat executat usant el optimitzador *Adam* i durant 50 *epochs*. A la *il·lustració 31* es mostra l'esquema del model seqüencial pel problema.



Il·lustració 31: Esquema del model seqüencial de la Xarxa Neuronal del problema de predicció de les col·lisions.

Un cop executat el model ens dona la següent matriu de confusió de la *il·lustració 32*.



Il·lustració 32: Matriu de confusió de l'algoritme de Xarxes Neuronals pel problema de predicció de les col·lisions.

L'anterior il·lustració destaca novament la segona diagonal de la matriu i es pot entreveure que la Xarxa Neuronal haurà estat efectiva en preveure correctament cada target, obtenint així una millor *Accuracy*.

Una detall interessant és que el model s'ha confós entre les instàncies de col·lisió lateral i frontó-lateral, potser degut a la seva similitud també a l'hora de donar-se. S'observa també que pels targets d'abast, col·lisió lateral i frontó-lateral s'han classificat moltes de les instàncies com si fossin d'aquestes classes. Veure les columnes de la matriu per aquests atributs. Això es creu pot ser degut a que aquests atributs siguin els més comuns del dataset i que per l'algoritme aquests hagin tingut un major pes.

El resultat del F1-Score es pot veure a la *Taula 12*.

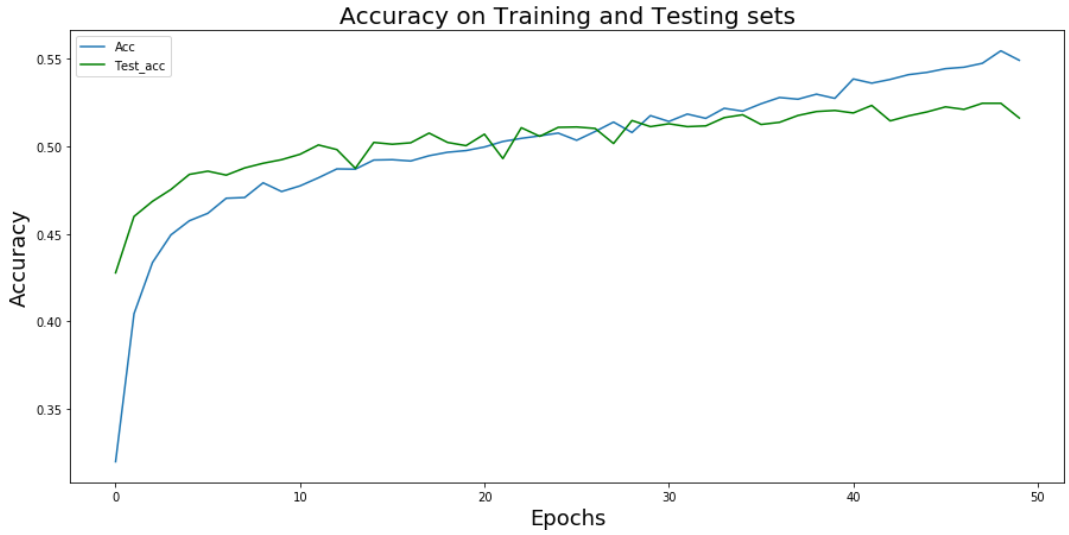
Taula 12: Mètriques de Precision, Recall i F1-Score pels diferents tipus de col·lisió per l'algoritme de Xarxa Neuronal.

	Precision	Recall	F1-Score	Support
Abast	0.63	0.22	0.33	1146
Abast múltiple	0.78	0.79	0.79	325
Atropellament	0.89	0.92	0.90	586
Caiguda 2 rodes	0.68	0.24	0.36	477
Caiguda Int. Veh.	0.85	0.69	0.76	144
Fronto-lateral	0.60	0.16	0.25	988
Lateral	0.55	0.21	0.30	1075
Element estàtic	0.85	0.19	0.31	145
Weighted Avg.	**0.67**	**0.34**	**0.42**	**4886**

Veiem que els resultats ponderats no han estat massa bons degut a que el *F1-Score* dels targets amb més pes no ha estat bona. Destaca en canvi la puntuació de 0.90 del atropellament que és el resultat de que només pot ser atropellat si és vianant.

Finalment s'ha aconseguit una *Accuracy* del **0.587404** i del **0.515964** sobre el set d'entrenament i de test, respectivament.

Finalment, veiem l'entrenament del classificador al llarg de els diferents *Epochs* a la *il·lustració 33*.



Il·lustració 33: Evolució de l'Accuracy sobre el Training i Testing set.

Després d'aproximadament 35 *Epochs* el model comença a patir una mica *d'Overfitting* ja que la *Accuracy* del *Testing set* no creix tan ràpid com la del *Testing set*. Tot i això el *l'Accuracy* del set d'entrenament continua creixent al llarg de les Epoch, passant del **0.42780** a un **0.515964.**

2.5.3.3 Discussió

Comparem els outputs de cada model per, un cop més, veure quin és el millor per cada problema.

Taula 13: Resum de l'Accuracy i el F1-Score sobre els dos algoritmes analitzats i classificador base.

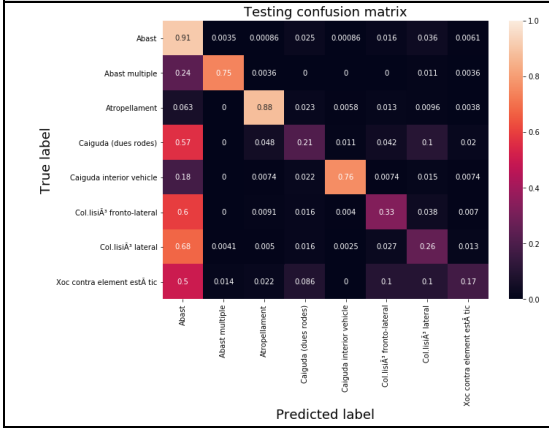
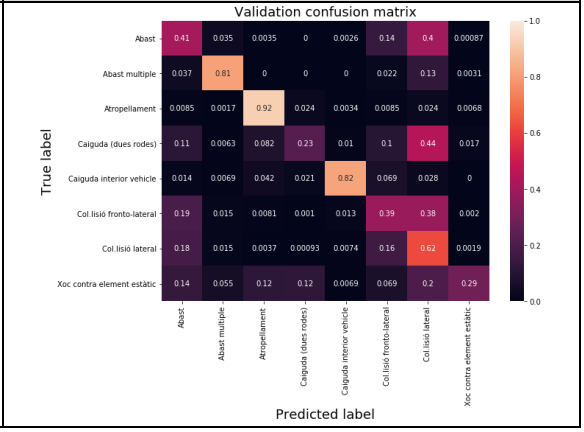
	Dummy Classifier		Random Forest		Xarxes Neuronals	
Accuracy	Δ -	**0.2330**	Δ 40,95%	**0.39460**	Δ 54,84%	**0.515964**
F1-Score Avg.		-	Δ 16%	**0.50000**	Δ -	**0.420000**

En aquest cas tant *l'Accuracy* com el F1-Score ens apunten que el millor algoritme a utilitzar en aquest problema és també el de les Xarxes Neuronals. Aquest algoritme ha aconseguit un resultat a la *Accuracy* un **23,5%** millor que el Random Forest i un **54,84%** millor que el *Dummy Classifier* que consistia en agafar la classe més comuna. Tot i els resultat, s'observa que ambdós classificadors han obtingut un resultat molt més bo que el del classificador base.

Pel que fa el *F1-Score* és millor per l'algoritme del Random Forest ja que classifica millor les instàncies amb més entrades però té molts *Falsos Positius* a l'atribut dels abasts. Però aquest **16%** amb un problema balancejat no és suficient per superar aquesta *Accuracy* un **23,5%** millor del model de les Xarxes Neuronals.

Si mirem el problema des del punt de vista de les matrius de confusió obtenim que també les Xarxes Neuronals obtenen un resultat on es defineix millor la diagonal de *True Possitives* per la majoria de targets. Encara que amb els targets amb més entrades com són l'abast i les col·lisions laterals i frontó-laterals l'algoritme es confon.

Taula 14: Comparativa de les matrius de confusió dels diferents algoritmes.

Random Forest	Xarxes Neuronals
	

Es conclou, per tant, que el model de les Xarxes Neuronals profundes ha estat millor en aquest problema degut a un bona Accuracy del **0.515964** tot i tenir un *F1-Score* pitjor que el Random Forest.

3. Conclusions

L'objectiu d'aquest treball ha estat el d'analitzar i predir tres tipus de problemes diferents de classificació binaria de múltiples classes. A diferència d'altres treballs relacionats, que s'han centrat majoritàriament en l'anàlisi només de ferits amb cotxes, aquest treball pot aportar més detall sobre com es produeixen els accidents de trànsit amb els seus diferents agents, models de cotxes i tipus de col·lisions.

L'anàlisi realitzat amb Tableau permet assegurar que (en l'entorn de les dades analitzades) un motorista té 5 vegades més probabilitats de partir un accident greu i 3 cops més probabilitats de patir un accident mortal que un cotxe. Així mateix, que els models i el color dels cotxes es poden relacionat amb l'edat dels conductors, i que en només 6 barris es concentren més del 30% dels accidents de la ciutat. Per últim, que el 50% dels accidents amb ferits i el 70% d'accidents amb víctimes mortals passen a menys de 10 metres de semàfors. Aquesta informació pot ser utilitzada per a la millora de la mobilitat de Barcelona, en la reducció dels accidents de trànsit o per a predir els models dels cotxes que més probablement es donin a la fuga després d'un accident. Aquest darrer element proporciona una eina que pot ser usada per la policia per reduir el ventall de possibilitats a l'hora de localitzar fugitius.

Des d'un punt de vista més tècnic, l'ús d'eines de Machine Learning pel problema de les classes descompensades, i en particular amb tècniques com *SMOTE-Tomek*, ha permès millorar les mètriques obtingudes. Així mateix, permet constatar que la mètrica del F1-Score també és important en l'anàlisi dels resultats. Pels problemes de classificació dels models de cotxe i després del tipus col·lisions, s'ha vist que l'algoritme amb una millor *Accuracy* ha estat les Xarxes Neuronals, que ha obtingut una *Accuracy* un 21% i 23,5% millor que el Random Forest en els dos últims problemes respectivament.

Finalment els resultats ens mostren que no hi ha un algoritme millor sempre per tots els problemes, sinó que s'ha de determinar amb l'ús de diferents mètriques el millor classificador per a cada cas.

4. Referències

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, *6*(1), 20. https://doi.org/10.1145/1007730.1007735

Breiman, L. (2001). Random Forests. *Kluwer Academic Publishers*, 157–175. https://doi.org/10.1007/9781441993267_5

Brownlee, J. (2018). Classification Accuracy is Not Enough : More Performance Measures You Can Use Your Start in Machine. Retrieved April 26, 2019, from https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/

Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron) - a review of applications in the atmospheric sciences. *Atmospheric Environment*, *32*(14–15), 2627–2636. https://doi.org/10.1016/S1352-2310(97)00447-0

Mordant, N., Delour, J., Lévêque, E., Arnéodo, A., & Pinton, J.-F. (2002). Long time correlations in {L}agrangian dynamics. *Advances in {T}urbulence {IX}*, *16*(1), 732–735. https://doi.org/10.1613/jair.953

Pozzolo, A. D., Caelen, O., & Johnson, R. A. (2019). *Featured Posts Credit Card Fraud Detection Analysis on Imbalanced Data - Part 1 The Plan Archive*. *2019*(March), 1–7.

RACC. (2018). Desplazamientos en moto en los accesos a Barcelona. Retrieved from RACC website: http://saladeprensa.racc.es/wp-content/uploads/2018/05/DOSSIER-RACC-Estudio-Motos-CAS.pdf

Tamerius, J. D., Zhou, X., Mantilla, R., & Greenfield-Huitt, T. (2016). Precipitation Effects on Motor Vehicle Crashes Vary by Space, Time, and Environmental Conditions. *Weather, Climate, and Society*, *8*(4), 399–407. https://doi.org/10.1175/wcas-d-16-0009.1

Theofilatos, A., Yannis, G., Kopelias, P., & Papadimitriou, F. (2016). Predicting Road Accidents: A Rare-events Modeling Approach. *Transportation Research Procedia*, *14*, 3399–3405. https://doi.org/10.1016/j.trpro.2016.05.293

WHO. (2018). WHO | Road traffic injuries. Retrieved from https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

Yuan, Z., Zhou, X., Yang, T., Tamerius, J., & Mantilla, R. (2017). Predicting Traffic Accidents Through Heterogeneous Urban Data : A Case Study. *Urban Computing*, 1–9. Retrieved from https://doi.org/10.475/123_4

5. Annexes

Totes les llibretes utilitzades per fer aquest treball es poden trobar a l'adreça https://github.com/jordisc97/TFG. Moltes de les línies de codi tenen comentaris per fer-les més explicatives. A continuació es descriurà breument que conté cada una d'aquestes.

1. **Persones – Accidents**

Primera llibreta que es va fer on es pot veure tot el llarg procés vist a l'apartat *2.2 Data Cleaning* amb dades de 2016 i 2017. Principalment en aquest dataframe es construeixen totes les Joins necessàries per ajuntar les dades dels accidents amb les condicions meteorològiques, sortides i postes de sol, els diferents càlculs, etc. Tot amb la finalitat de tenir les dades llestes pel procés d'anàlisi i Machine Learning pel problema de la classificació dels diferents tipus de ferits.

2. **Def – RF Tipus ferits**

A aquesta llibreta ja es pot trobar el primer algoritme de Machine Learning usat per la predicció, aquest és el Random Forest. En aquest codi es fa primerament un canvi de nom als targets de ferits lleus, greus i morts per valors numèrics del 0 al 2.

Seguidament es realitza la separació del dataframe a una part d'entrenament (75%) i les resta com a test. Es normalitzen els dades i s'aplica un Principal Component Analysis o PCA per veure si es pot treure alguns atributs sense perdre massa variància a les dades. El resultat d'aquest en tots els problemes és que no canviaria massa el temps d'execució, així que tots els atributs del dataframe s'insereixen als algoritmes.

A continuació s'aplica la tècnica de SMOTE-Tomek per tenir el mateix nombre d'entrades de cada classe. Finalment es passen les dades pel Random Forest que fa ús dels paràmetres marcats al Parameter Grid.

Un cop s'ha executat l'algoritme obtenim les mètriques de *Precision*, *Recall* i *F1-Score.* També s'obté la importància relativa de cada atribut amb un gràfic com el de la *il·lustració 20*. Finalment s'obtenen les mètriques *d'Accuracy* i la matriu de confusió.

3. **Def- ANN Tipus ferits**

En aquesta llibreta es trobat tot el procediment usat per convertir les dades que venien de la primera llibreta seguint l'apartat *2.3 Data Processing*. Es segueix el mateix procediment que pel Random Forest separant les dades en dos sets, *Training (75%)* i *Testing*. Seguidament es construeix el model seqüencial explicat a la *il·lustració 22* i es disposa a executar-lo durant unes 50 *Epochs*.

Obtenim també les mètriques del *F1-Score*, *l'Accuracy*, la matriu de confusió i l'evolució de *l'Accuracy* amb les *Epochs*.

4. **Def- RF Models**

Mateix procediment que pel Random Forest de ferits. Amb la única diferència de que les dades venen de la llibreta de Xarxes Neuronals per aquest mateix problema (següent apartat). Un cop executat l'algoritme s'obtindran les mètriques de *Precision*, *Recall* i *F1-Score,* a més de la matriu de confusió del problema.

5. **Def. ANN Predict the escaped car**

Aquesta llibreta conté les dades que usa la llibreta pel mateix problema però amb l'algoritme de Random Forest. Degut a que per cada codi d'accident només hi ha un únic tipus de col·lisió (col·lisió frontó-lateral, abast, etc.) es va poder ajuntar aquesta dada amb el dataframe dels models i tipus de cotxe.

A continuació es van seleccionar els 200 models amb més accidents de Barcelona. Es van considerar només els models de cotxe ja que entre els diferent models de motocicleta no hi ha tanta diferència.

Seguidament es van assignar el numero de portes per cada model i el quin tipus de cotxe, com per exemple si aquest era un 4x4, esportiu, familiar, etc. Aquestes dades van ser composades manualment fent ús de les dades proporcionades pels fabricants dels models o bé amb imatges de Google. Tot aquest procés es va fer pensant quines serien les característiques que recordaria la persona conductora del cotxe amb qui acaba de xocar.

També es va generar l'algoritme que ens serviria de classificador base que sabent la marca de cotxe de l'accident, responia amb el model de cotxe més comú per aquella marca.

Finalment les dades, van ser separades en dos sets i normalitzades. Es va construir el model seqüencial de la *il·lustració 27* i després de 200 *Epoch* es van aconseguir els resultats exposats a la pàgina 33.

6. **Def- RF Col·lisions**

En aquesta llibreta s'usen les dades obtingudes de tot els procés fet a la llibreta de Persones – Accidents i s'afegeixen els col·lisions ja que com s'ha dit anteriorment, els tipus de col·lisions són únics per cada codi d'accident. D'aquesta llibreta s'obté el classificador base per aquest problema fen ús del prior, és a dir, la classe més comuna serà la que sempre es donarà com a output per qualsevol entrada.

Un cop executat l'algoritme s'obtenen per aquest problema les mètriques de *Precision*, *Recall* i *F1-Score,* a més de la matriu de confusió.

7. **Def- ANN Col·lisions**

Per últim, s'usen les mateixes dades generades al Random Forest d'aquest problema per entrenar el model seqüencial de la *il·lustració 31* durant 50 *Epochs*. Seguint el mateix procediment que per les altres llibretes, s'obtenen les mètriques *d'Accuracy*, *F1-Score*, la matriu de confusió i les *Accuracies* en funció de les *Epochs*.