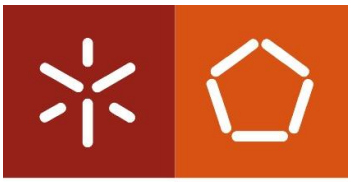

Universidade do Minho

Departamento de Informática

Dados e Aprendizagem Automática
Trabalho Prático de Grupo
Conceção e Otimização de Modelos de Machine Learning

Grupo Nº 22

José Martins (PG47369)
Maria Marques (PG47489)
Pedro Pereira (PG47581)
Pedro Pereira (PG47584)

16 de fevereiro de 2022

Conteúdo

Capítulo 1

Introdução

A realização deste trabalho prático tinha como principal objetivo envolver a equipa de desenvolvimento na área de Machine Learning, de forma a que fossem aplicados os conceitos abordados nas aulas teóricas e práticas da unidade curricular de Dados e Aprendizagem Automática. Foram propostos dois momentos de avaliação, um primeiro no qual foi fornecido pelos docentes um dataset no domínio do fluxo de tráfego rodoviário e um outro para o qual a equipa de desenvolvimento teve de escolher um dataset num domínio arbitrário que considerava mais interessante. Portanto, para este segundo momento foi escolhido um dataset relacionado com a meteorologia australiana.

Como consequência, cada secção deste relatório está divido em duas partes, uma para cada um dos conjuntos de dados. Em cada uma destas partes, será explicado o dataset em questão seguido da metodologia seguida para a sua preparação e tratamento. Serão também explicados e detalhados os modelos escolhidos e desenvolvidos bem como será feita uma análise dos resultados obtidos. Por fim serão elaboradas conclusões baseadas nesses mesmos resultados.

É também importante mencionar que para cada um dos modelos desenvolvidos, foi estabelecido como objetivo principal obter a melhor precisão possível através da aplicação de variadas técnicas aprendidas.

Capítulo 2

Descrição do conjunto de dados

2.1 Dataset da previsão do fluxo de tráfego rodoviário

2.1.1 Descrição do dataset

Tal como mencionado anteriormente, este dataset foi-nos fornecido pela equipa docente. Este contém dados referentes ao tráfego de veículos na cidade do Porto durante um período superior a 1 ano. O objetivo definido era o de desenvolver modelos de Machine Learning capazes de prever o fluxo de tráfego rodoviário, numa determinada hora, na referida cidade. De seguida enumeram-se e especificam-se os atributos que constituem o conjunto de dados em questão:

- **city_name** - nome da cidade em causa;
- **record_date** - o timestamp associado ao registo;
- **average_speed_diff** - a diferença de velocidade corresponde à diferença entre a velocidade máxima que os carros podem atingir em cenários sem trânsito e a velocidade que realmente se verifica. Quanto mais alto o valor, maior é a diferença entre o que se está a andar no momento e o que se deveria estar a andar sem trânsito, i.e., valores altos deste atributo implicam que se está a andar mais devagar;
- **average_free_flow_speed** - o valor médio da velocidade máxima que os carros podem atingir em cenários sem trânsito;
- **average_time_diff** - o valor médio da diferença do tempo que se demora a percorrer um determinado conjunto de ruas. Quanto mais alto o valor maior é a diferença entre o tempo que demora para se percorrer as ruas e o que se deveria demorar sem trânsito, i.e., valores altos implicam que se está a demorar mais tempo a atravessar o conjunto de ruas;
- **average_free_flow_time** - o valor médio do tempo que demora a percorrer um determinado conjunto de ruas quando não há trânsito;

- **luminosity** - o nível de luminosidade que se verificava na cidade do Porto;
- **average_temperature** - o valor médio da temperatura para o *record_date* na cidade do Porto;
- **average_atmosp_pressure** - o valor médio da pressão atmosférica para o *record_date*;
- **average_humidity** - o valor médio da humidade para o *record_date*;
- **average_wind_speed** - o valor médio da velocidade do vento para o *record_date*;
- **average_cloudiness** - o valor médio da percentagem de nuvens para o *record_date*;
- **average_precipitation** - o valor médio de precipitação para o *record_date*;
- **average_rain** - avaliação qualitativa da precipitação para o *record_date*.

É importante referir que foi também disponibilizado um dataset de teste, utilizado para validar a accuracy do modelo de dados na plataforma Kaggle. O objetivo principal desta fase era o de prever, para cada registo do dataset de teste, o nível de trânsito correspondente.

2.1.2 Tratamento dos dados

Devido à importância desta fase, o grupo começou por realizar alguma análise do dataset, discutindo entre os elementos quais as técnicas que seriam adequadas para tratar os dados em causa e só depois procedeu à implementação das técnicas consideradas adequadas.

Em primeiro lugar, foi decidido que o atributo *city_name* nada contribuía para a previsão do trânsito, pelo que este apenas continha um único valor. Por este motivo esta coluna foi removida tanto no dataset de treino como no de teste. Da mesma forma e pelo mesmo motivo foi também removido o atributo *average_precipitation*, que apenas continha um único valor de 0 para todas as linhas.

De seguida, procedeu-se ao tratamento dos valores duplicados, removendo-se também todos os valores que se encaixavam nessa restrição.

Passou-se então ao tratamento de *outliers*. Através da análise dos dados, mencionada posteriormente, neste caso através da observação de cada *boxplot*, foram identificados atributos que continham um grande número de *outliers*, pelo que foi necessário resolver esse problema. A solução baseou-se numa técnica de remoção de outliers que consiste em substituir os valores desse tipo pelos valores dos limites dos *boxplots*.

Na análise dos dados foi também denotado que existiam em algumas colunas valores diferentes que apresentavam o mesmo significado. Por isso, para tais valores foi decidido que seriam todos considerados como um mesmo valor procedendo-se a um conjunto de substituições que serão melhor especificadas já de seguida. Um dos atributos que possuía esse problema era o *average_cloudiness* que apresentava valores como "nuvens quebrados"e "nuvens quebradas", que significam claramente o mesmo, bem como "tempo nublado"e "nublado", "céu

claro"e "céu limpo", "nuvens dispersas"e "algumas nuvens". Todos estas semelhanças foram resolvidas com a atribuição de um mesmo valor.

No passo seguinte recorreu-se ao *Feature Engineering* das datas de recolha dos dados, uma vez que não eram totalmente importantes para a previsão do trânsito, ao contrário da hora, que poderá corresponder a horas de ponta e horas específicas em que o trânsito é elevado. Por esse motivo, decidiu-se extrair a hora a partir da data, criando um novo atributo *hour*. De seguida removeu-se a coluna de *record_date* uma vez que já não seria útil.

A análise de dados feita inicialmente também permitiu perceber que o atributo *average_rain* possuía cerca de 80% dos seus valores como nulos, pelo que não valeria a pena estar a prever esses valores com apenas 20% dos dados, uma vez que poderia ser prejudicial aos resultados finais obtidos. Tendo isto em conta, este atributo foi também removido.

Era necessário também tratar os atributos categóricos como *luminosity*, *average_cloudiness* e *average_speed_diff*. No caso das duas primeiras apenas foi necessário passar os valores categóricos para numéricos. Porém, no caso do terceiro atributo, o grupo recorreu à técnica de *One-Hot-Encoding*, uma vez que sendo a variável dependente seria mais benéfico para a análise dos resultados finais ter uma coluna para cada um dos valores.

Por último foi preciso tratar dos valores nulos. Como neste ponto apenas o atributo *average_cloudiness* possuía valores nulos, apenas a este foi aplicado o tratamento. A equipa decidiu que os valores nulos deviam ser recalculados e substituídos. Para o fazer foi utilizado um método de SVC, que a partir dos valores não nulos da coluna e dos valores de colunas que têm influência sobre esse atributo, permitiu prever os valores nulos da coluna em questão. Este método foi auxiliado por uma GridSearch que permitiu calcular os melhores parâmetros do modelo para prever esses valores.

2.1.3 Conceção e otimização de modelos

De forma a atingir o objetivo final de prever os valores de *average_speed_diff* a equipa desenvolveu um modelo de Redes Neuronais. De salientar que antes do uso deste modelo outros foram experimentados e testados, tendo-se verificado que esta seria a técnica que permitiria atingir melhores resultados.

O primeiro passo para a utilização da rede foi o de escalar os dados, uma vez que redes neuronais necessitam de dados escalados com valores entre 0 e 1. De seguida passou-se para a implementação do modelo de Rede Neuronal propriamente dito. Tal como sabemos uma rede neuronal é divida em camadas, camada de input, camadas escondidas e camadas de output. Assim para cada uma destas foram definidos o número de neurónios que achámos adequados para construir a rede. O número de neurónios na camada de input foi de 11, correspondendo ao número de atributos que constituíam o dataset de treino. Nas camadas escondidas, foram dados números de neurónios baseados naquilo que se achava ser um número suficiente para tornar o modelo eficiente. No caso da camada de output o número de neurónios estabelecido foi de 5 baseado nos possíveis valores que podem ser obtidos como resultado na previsão da *average_speed_diff*. Quanto à função de ativação da rede neuronal, uma vez que os valores que se querem prever são do tipo categórico, recorreu-se a uma função de Softmax. Esta função, de forma semelhante à bem conhecida função Sigmoid, calcula as probabilidades relativas dos valores, para posteriormente se definirem

a que classe pertencem. Em seguida, afinámos o nosso modelo, utilizando uma GridSearch de forma a achar os melhores hiperparâmetros, tanto para a função de ativação como para a *learning rate*. Foram ainda definidas a *"loss function"* (neste caso, CategoricalCrossentropy, uma vez que o valor a prever é do tipo categórico), o *"optimizer"* (Adam com uma *learning rate* de 0.01, responsável por implementar o gradiente descendente e por atualizar os pesos nos neurónios) e a métrica *CategoricalAccuracy*. De salientar que todos estes campos foram modificados de forma a resultar no melhor resultado possível.

2.2 Dataset da previsão de chuva na Austrália

2.2.1 Descrição do dataset

O dataset escolhido pelo grupo, Rain in Australia, lida com a previsão de chuva para o dia seguinte. Este conjunto de dados contém cerca de 10 anos de observações meteorológicas diárias de muitos locais da Austrália. Contamos com 145460 registos que podem conter valores nulos/inexistentes. Deste modo, ao todo existem 23 atributos diferentes:

- **Date** - Data da recolha dos dados, sob a forma de yyyy/mm/dd. Não contém valores nulos.
- **Location** - Local onde os dados foram recolhidos.
- **MinTemp** - Temperatura mínima registada naquele dia em graus celcius, 1485 dados nulos (1%).
- **MaxTemp** - Temperatura máxima registada naquele dia graus celcius, com um total de 1261 valores nulos (1%).
- **Rainfall** - A quantidade de chuva registada para o dia em mm, 3261 valores nulos (2%).
- **Evaporation** - A chamada evaporação do tanque Classe A (mm) nas 24 horas às 9h. Com um total de 62790 registos nulos (43%).
- **Sunshine** - O número de horas de sol durante o dia, com 69835 valores nulos (48%).
- **WindGustDir** - A direção da rajada de vento mais forte nas 24 horas à meia-noite, 10326 valores nulos (7%)
- **WindGustSpeed** - A velocidade (km / h) da rajada de vento mais forte das 24 horas à meia-noite, 10263 valores nulos (7%).
- **WindDir9am** - Direção do vento às 9h, 10566 valores nulos (7%).
- **WindDir3pm** - Direção do vento às 15h, com cerca de 4228 valores nulos (3%).
- **WindSpeed9am** - Velocidade do vento (km / h) em média mais de 10 minutos antes das 9h, 1767 valores nulos correspondentes a (1%).

- **WindSpeed3pm** - Velocidade do vento (km / h) em média mais de 10 minutos antes das 15h, 3062 valores nulos (2%).
- **Humidity9am** - Humidade (percentagem) às 9h, 2654 valores nulos (2%).
- **Humidity3pm** - Humidade (percentagem) às 15h, 4507 valores nulos (3%).
- **Pressure9am** - A pressão atmosférica (hpa) foi reduzida para o nível médio do mar às 9h, 15065 valores nulos (10%).
- **Pressure3pm** - A pressão atmosférica (hpa) foi reduzida para o nível médio do mar às 15h, 15028 valores nulos (10%).
- **Cloud9am** - Fração de céu obscurecida por nuvem às 9h. Isso é medido em "octas", que são uma unidade de oitavos. Com 55888 valores nulos (38%).
- **Cloud3pm** - Fração de céu obscurecida por nuvem às 15h. Isso é medido em "octas", que são uma unidade de oitavos. Contem 59358 valores nulos ou seja (41%).
- **Temp9am** - Temperatura (graus C) às 9h, 1767 valores nulos (1%).
- **Temp3pm** - Temperatura (graus C) às 15h, 3609 valores nulos (2%).
- **RainToday** - Booleano: 1 se a precipitação (mm) nas 24 horas às 9h exceder 1 mm, caso contrário 0, com 3261 valores nulos (2%).
- **RainTomorrow** - A quantidade de chuva no dia seguinte em mm. Usado para criar a variável de resposta RainTomorrow. Uma espécie de medida do "risco". Contém 3267 valores nulos (2%).

2.2.2 Tratamento dos dados

Analisando o dataset, conclui-se que a maioria dos atributos tem uma percentagem de valores nulos inferior a 7%, à exceção dos atributos *Evaporation*, *Sunshine*, *Cloud9am*, *Cloud3pm* que, em contraste com os restantes, possuem respetivamente uma percentagem de 43%, 48%, 38% e 41% de valores nulos. Se nos limitássemos a substituí-los pelo valor mais frequente, enviesaria os resultados e as conclusões a que pudéssemos chegar. Assim para a análise do problema, decidiu-se eliminá-los. Dos 23 atributos iniciais, estamos, então, a considerar 19 atributos. Outro tratamento que feito, foi relativamente ao atributo Date, ao qual através do processo de *feature engineering* alterámos o formato da mesma, criando novos atributos associados aos dias, meses e anos e removendo o formato original. Os seguintes passos de preparação do dataset consistiram no tratamento de valores nulos e de *outliers*. Para o tratamento dos valores nulos sabemos que há diferentes quantidades dos mesmo conforme o atributo observado. Assim dependendo de sua distribuição, substituímos esses valores ora pela mediana ora pela moda, sendo que para todos os atributos categóricos foram substituídos os valores nulos pela moda correspondente, enquanto que para todos os atributos contínuos estes foram substituídos pela mediana respetiva.

Quanto ao tratamento de outliers utilizou-se a mesma estratégia referida na explicação do dataset anterior, ou seja, os outliers presentes em cada atributos foram substituídos pelos valores dos seus limites superiores ou inferiores conforme o caso.

Foi ainda feita a conversão dos atributos categóricos (*Location*, *WindDir9am*, *WindDir3pm*, *WindGustDir*) em valores numéricos através do método LabelEncoder. Os atributos *RainTomorrow* e *RainToday* foram também transformados em valores numéricos, onde os valores 'Yes' passaram a corresponder ao valor 1 e os valores 'No' ao valor 0.

Por fim, verificámos ainda que os dados do nosso dataset estavam extremamente desbalanceados.

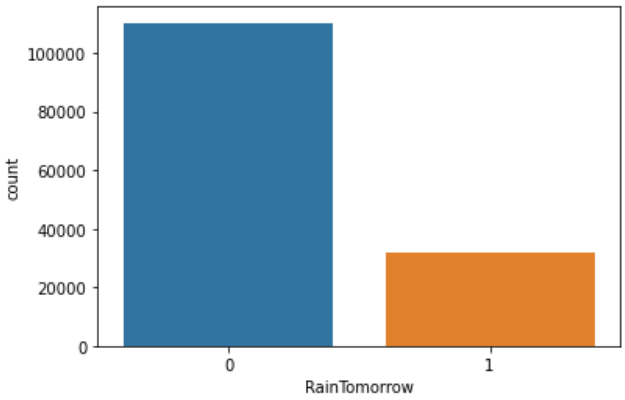

Figura 2.1: Gráfico de barras

Desta forma antes de passar para a conceção do modelo foi de extrema importância balancear o dataset. Para tal utilizámos a biblioteca imblearn a qual contém a classe RandomUnderSampler. Esta permitiu-nos utilizar a técnica de undersampling, que para equilibrar o conjunto de dados desiguais, manteve todos os dados na classe minoritária, que neste caso corresponde ao 1 do atributo *RainTomorrow*, e diminuiu o tamanho da classe maioritária, valor 0. Desta maneira foi possível extrair informação mais precisa do conjunto de dados originalmente desequilibrado.

Após estes tratamentos os dados estão limpos e prontos para a fase de modelagem.

2.2.3 Conceção e otimização de modelos

De forma a atingir o objetivo final de prever os valores de *RainTomorrow* a equipa desenvolveu um modelo de Redes Neuronais. De salientar que antes do uso deste modelo outros foram experimentados e testados, tendo-se verificado que esta seria a técnica que permitiria atingir melhores resultados.

O primeiro passo para a implementação da rede foi o de escalar os dados, uma vez que redes neuronais necessitam de dados escalados com valores entre 0 e 1.

De seguida passou-se para a implementação do modelo de rede Neuronal propriamente dito. Tal como sabemos uma rede neuronal é divida em camadas,

camada de input, camadas escondidas e camadas de output. Assim para cada uma destas foram definidos o número de neurónios que achámos adequados para construir a rede. O número de neurónios na camada de input foi de 20, correspondendo ao número de atributos que constituíam o dataset de treino. Nas camadas escondidas, foram dados números de neurónios baseados naquilo que se achava ser um número suficiente para tornar o modelo eficiente. No caso da camada de output o número de neurónios estabelecido foi de 1. Quanto à função de ativação da rede neuronal foi utilizada a função sigmoid, de forma a obter um valor entre 0 e 1.

Depois de construir o modelo, foi necessário definir a *"loss function"* (neste caso, *binary_crossentropy*), o *"optimizer"* (Adam com uma *learning rate* de 0.0001, responsável por implementar o gradiente descendente e por atualizar os pesos nos neurónios) e o conjunto de métricas *"accuracy"*. De salientar que todos estes campos foram modificados de forma a resultar no melhor resultado possível.

Capítulo 3

Resultados Obtidos

Esta secção do documento visa apresentar os resultados obtidos através dos modelos construídos que serão acompanhados por uma análise crítica dos mesmos, bem como por um conjunto de gráficos que permitirão visualizar o comportamento dos modelos.

3.1 Obtenção e análise crítica de resultados

3.1.1 Dataset da previsão do fluxo de tráfego rodoviário

Uma vez executado o nosso modelo de rede neuronal, anteriormente programado e especificado, chegou a altura de analisar os resultados obtidos.

O primeiro passo consistiu na utilização de um gráfico que nos permitiu identificar se o nosso modelo originou *Overfitting*. O gráfico em questão é apresentado abaixo e através dele podemos inferir que não existe um caso de *Overfitting*.

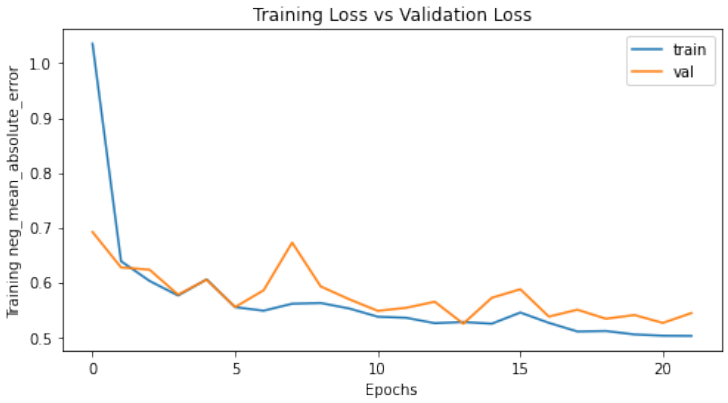

Figura 3.1: Gráfico de Overfitting

De seguida, utilizou-se uma matriz de confusão (tipo de gráfico normalmente usado para analisar resultados provenientes de redes neuronais). Esta é apresentada abaixo, e nela podemos observar uma diagonal bem definida, o que

significa que os valores previstos são na sua generalidade idênticos aos valores de teste, sendo observados valores razoáveis de sensibilidade e precisão.

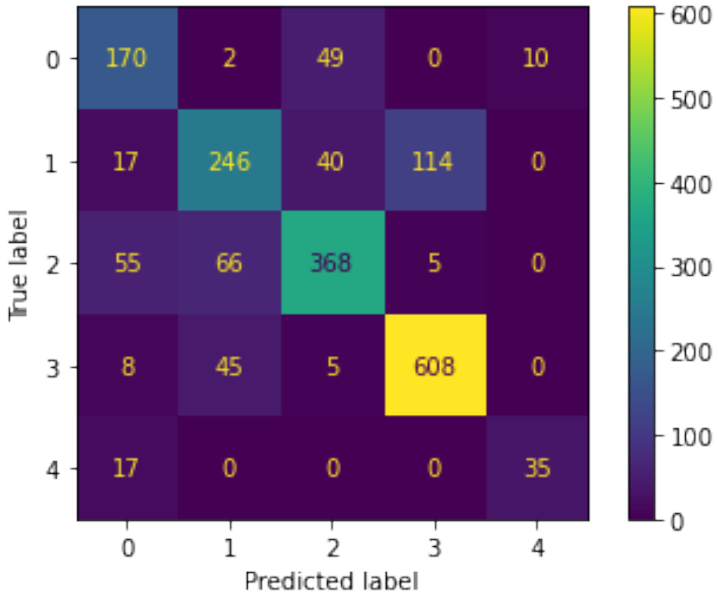

Figura 3.2: Matriz de confusão

Finalmente, calculou-se o valor da *accuracy* atingido pelo modelo de treino:**0.7812**.

3.1.2 Dataset da previsão de chuva na Austrália

A fase de análise de resultados deste dataset seguiu um processo muito semelhante ao anterior.

Tal como no dataset mencionado acima, o primeiro passo consistiu na utilização de um gráfico que nos permitiu identificar se o nosso modelo originou *Overfitting*. O gráfico em questão é apresentado abaixo e através dele podemos inferir que não existe um caso de *Overfitting*.

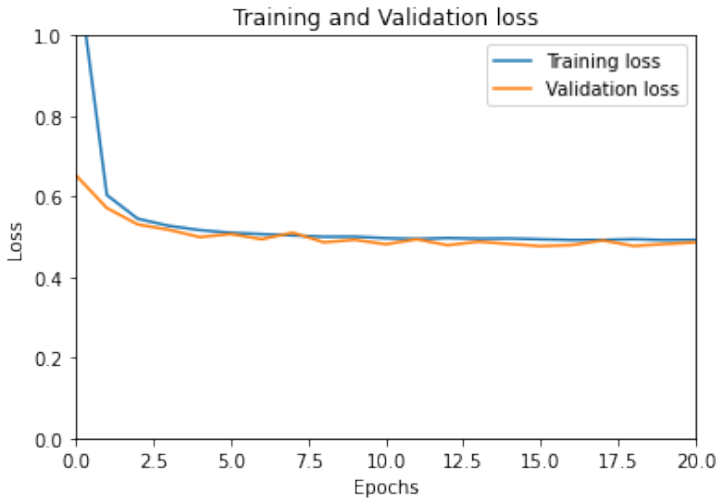

Figura 3.3: Gráfico de Overfitting

De seguida, utilizou-se uma matriz de confusão (tipo de gráfico normalmente usado para analisar resultados provenientes de redes neuronais). Esta é apresentada abaixo, e nela podemos observar uma diagonal bem definida, o que significa que os valores previstos são na sua generalidade idênticos aos valores de teste, sendo observados valores razoáveis de sensibilidade e precisão, sendo estes 0.75 e 0.79 respetivamente.

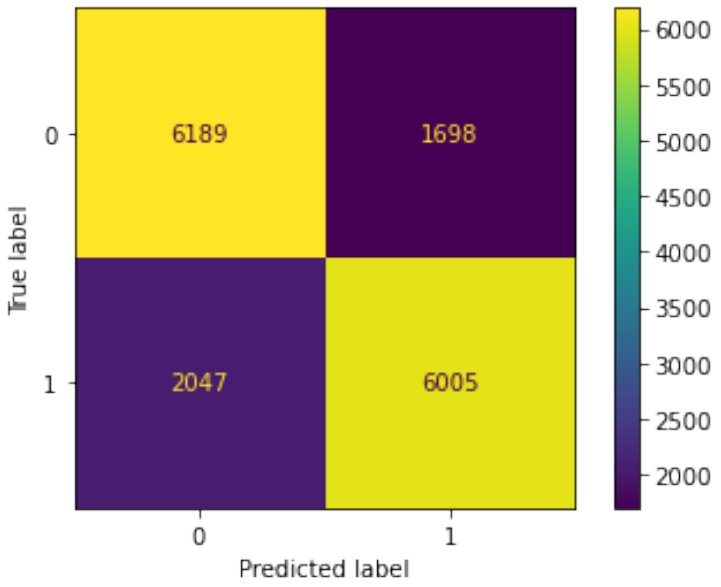

Figura 3.4: Matriz de confusão

Finalmente, calculou-se o valor de *accuracy* atingido pelo modelo de treino: **0.77**.

3.2 Apresentação de sugestões e recomendações

Após a análise dos resultados obtidos e dos modelos desenvolvidos, a equipa de desenvolvimento discutiu entre os seus elementos o que poderia ser ser melhorado no futuro relativamente ao desenvolvimento deste projeto. Concluiu-se assim que talvez pudessem ser aplicadas diferentes técnicas de tratamento de dados.

Para além disso, de forma a melhorar a eficácia dos modelos desenvolvidos, poderiam ser testadas diferentes estratégias quanto à organização das redes neuronais, uma vez que existem imensas possibilidades que não puderam ser testadas, e que com o poder da aprendizagem automática através de redes neuronais certamente serão possíveis resultados ainda mais positivos.

Capítulo 4

Conclusões

O grupo apresenta-se satisfeito com o trabalho desenvolvido, embora seja claro que este ainda poderia ser melhorado, principalmente ao nível do tratamento de dados.

Este trabalho despertou algum interesse nos elementos do grupo, por se tratar de um projeto que lida com dados e problemas reais. Este permitiu perceber que a área de Machine Learning tem imensas aplicações no mundo real.

A realização deste trabalho permitiu ao grupo não só aplicar os conceitos que adquiriu ao longo de todo o semestre, mas também perceber o verdadeiro funcionamento de todos os modelos que foi aplicando ao longo do desenvolvimento do trabalho.