SISTEMAS INTELIGENTES PARA LA GESTIÓN EN LA EMPRESA

MASTER PROFESIONAL EN INGENIERÍA EN INFORMÁTICA

Práctica 2: Multiclasificación

Autores
José Ángel Díaz García y Pablo Martin-Moreno

Equipo
José Ángel & Pablo

Puntuación y Posición
0.82297

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Granada, Junio de 2017

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

Esta última práctica está enmarcada dentro de la asignatura **Sistemas Inteligentes para La Gestión en la Empresa** del Master Profesional en Ingeniería Informática de la UGR y aborda un problema real de predicción multiclase en la plataforma Kaggle [2].

Este problema, es de un nivel avanzado, y a lo largo de los siguientes capítulos intentaremos aportar una solución aceptable en la plataforma Kaggle, así como estudiar y asentar los diferentes conceptos teóricos vistos en la asignatura.

1.1. Problema y Dataset

El problema en última instancia es un problema de clasificación multiclase real el cual deberá ser resuelto mediante técnicas de *deeplearning*. El problema, en concreto es Intel & MobileODT Cervical Cancer Screening [3] y trata de clasificar partiendo de imágenes del cervix de distintas pacientes, que tipo de tratamiento para el cancer es más efectivo, aspecto muy relevante sobre todo en puntos del mundo rural donde el acceso a grandes infraestructuras médicas puede estar limitado y donde la prevención en etapas tempranas puede ser decisiva.

El dataset está compuesto de la siguiente manera:

- **test**: 512 Imágenes que deberemos clasificar tras el entrenamiento.

- **train**: Tenemos un total de 1581 muestras para entrenar, compuestas por 350 de tipo 1, 781 de tipo 2 y 450 de tipo 3.
- **train-extra**: El dataset ofrece también una gran conjunto de imagenes extra para el entrenamiento de unos 30GB de espacio en disco.

Podemos ver como el dataset muestra cierto ratio de des balanceo, por lo que en instancias superiores de la práctica, deberemos atacar este punto para obtener mejores resultados. Por otro lado, las restricciones del problema hacen que sean interesantes propuestas como *One Vs One* o *One vs All* que analizáremos en sucesivos puntos.

1.1.1. Evaluación

La evaluación de está práctica tendrá como evaluador del modelo la función *logloss*, donde los falsos negativos tendrán una gran penalización. En caso de obtener una evaluación perfecta, el *logloss* del clasificador sería **0**.

1.2. Herramientas y objetivos

En esta sección veremos una breve introducción a las herramientas usadas para el desarrollo de la práctica así como de los principales objetivos que se buscan conseguir con el desarrollo de la misma.

1.2.1. Hardware

Elemento	Características
Procesador	2,6 GHz Intel Core i5
GPU	-
Memoria Ram	8 GB 1600 MHz DDR3
Disco duro	SATA SSD de 120 GB

Tabla 1.1: Especificaciones técnicas de la máquina 1.

Elemento	Características
Procesador	Intel Core i7 6700HQ
GPU	2,6 GHz Intel Core i5
Memoria Ram	16 GB SDRAM
Disco duro	128GB SSD

Tabla 1.2: Especificaciones técnicas de la máquina 2.

1.2.2. Software

El software utilizado es en su práctica totalidad software libre, siendo el restante software propietario cuyas licencias vienen incluidas en el sistema operativo de la máquina 1.1 siendo este OS X "Sierra", o el Windows 10 de la máquina 1.2. El software usado es:

- **RStudio**: Entorno de trabajo para R.
- **Tensorflow**: Entorno de deeplearning sobre Python.
- **Keras**: Capa de abstracción sobre Tensorflow.
- **MXNet**: Libreria de Deeplearning sobre R.
- **Atom**: Editor de texto plano para la programación de los scripts.
- **TeXShop**: procesador de textos basado en *Latex* usado para elaborar la documentación del presente proyecto.

1.2.3. Objetivos

Los objetivos de este trabajo podrían resumirse en los siguientes:

- Obtener un modelo predictivo fiable que dado una nueva imagen pueda predecir el tipo de tratamiento contra el cancer a aplicar.
- Obtener un valor de *LogLoss* aceptable para escalar posiciones en la competición de Kaggle.

- Comprender y estudiar las distintas técnicas de minería de datos vistas en la asignatura.
- Ahondar en el proceso de la multiclasificación y las vertientes de estudio dentro de la misma.
- Estudiar distintos métodos de clasificación sobre el mapa de características proveniente del entrenamiento de las redes neuronales.

1.3. Organización del trabajo

La organización del presente documento, se centra en detallar cada uno de los pasos seguidos durante el estudio y resolución del problema planteado en esta introducción. En el capítulo 2 veremos los scripts y explicaciones asociadas al preprocesado de datos, más concretamente al data augmentation y cierto enfoque de *imbalance learning*. En el capitulo 3 tendremos el grueso del trabajo en el que entraremos en detalle en una primera aproximación *from scracth* con *mxnet* y el grueso del trabajo realizando *fine tunning* en Tensorflow. Finalizaremos con el capítulo 4 dedicado al estudio de multiclasificación y uso de otros clasificadores y por último las conclusiones y vías futuras que quedan relegadas al capítulo 5.

Los resultados obtenidos en la competiciÓn de kaggle pueden encontrarse en el Anexo de la seccion 6.

Capítulo 2

Preprocesado

2.1. Data Augmentation

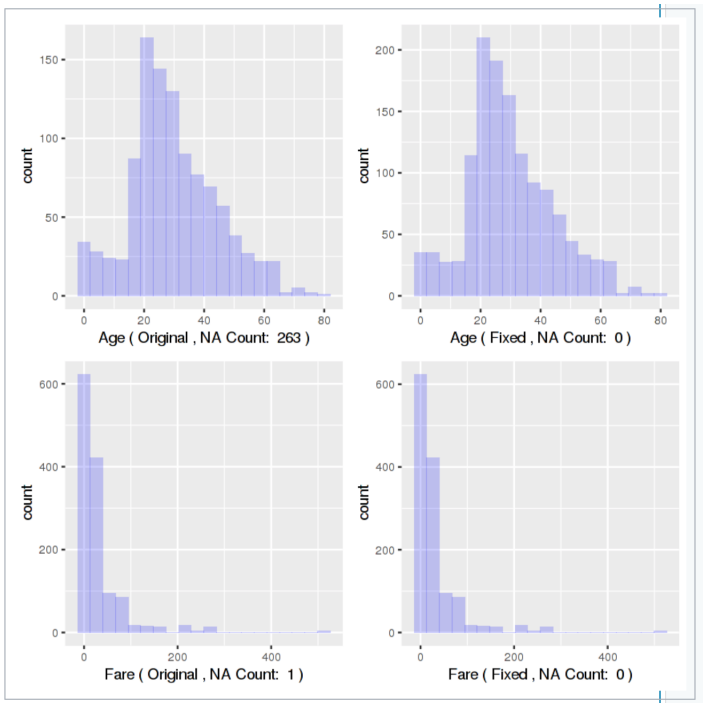

Figura 2.1: Comparación de la imputación de valores perdidos.

2.2. Imbalance learning

Capítulo 3

Clasificación con NN

3.1. From Scratch

3.2. Fine Tunning

3.3. Transfer Learning

Capítulo 4

Multiclasificación y mapas de características

4.1. Multiclasificacion

4.2. Mapas de características

Capítulo 5

Conclusiones y vías futuras

En este capítulo final se estudian los resultados obtenidos a lo largo del trabajo y vías futuras para aumentar aún más el accuracy. También se complementan las conclusiones que se han ido obteniendo a lo largo del trabajo. Por tanto, podríamos resumir las conclusiones finales del trabajo en las siguientes:

5.1. Conclusiones finales

5.2. Vías futuras

Capítulo 6

ANEXO I

En este Anexo, podemos encontrar la tabla con los resultados obtenidos en las distintas entregas a Kaggle, el número de resultados no coincide al 100 % con los subidos a Kaggle, dado que a petición del profesor de teoría, Francisco Herrera, se realizó una batería de experimentos sobre el problema con el algoritmo XGBoost. Los resultados de estos experimentos fueron obviados del problema ya que no ofrecían mejora alguna sobre los resultados.

Sol	***Preprocesado***	***Algoritmos***	***Acc Test***	***Acc Training***	***Pos***
1	Imputados valores perdidos	Asumimos todos mueren	0.61	0.616	6429
2	Imputados valores perdidos	Asumimos que solo viven las mujeres	0.7655	0.7867	4781
3	Imputados valores perdidos	Todos los hombres y las mujeres de tercera clase con >20 de Fare mueren, las demás viven.	0.77990	0.8080	3259
4	Imputados valores perdidos Creada variable isChild	Ademas de lo anterior, niños de 2 clase viven.	0.78240	0.8181	2877
5	Imputados valores perdidos	Random Forest nativo de R	0.78469	0.8316	2543
6	Imputados valores perdidos añadidas variable Title, isMother, isChild	Random Forest nativo de R	0.78469	0.8316	2543
7	Imputados valores perdidos, añadida variable Title	Random Forest rparty	0.79904	0.838	1732
8	Imputados valores perdidos, añadida variable Title, eliminados outliers	Random Forest rparty	0.80383	0.8432	936
9	Imputados valores perdidos, añadidas title, familysize y familyID,	Random Forest rparty	0.8134	0.8356	473
10	Añadidas nuevas características Eliminado outliers, imputados valores perdidos	Random Forest muy ajustado y probado con distintos parámetros quedándonos con el mejor en training	0.82297	0.88431	218

Tabla 6.1: Tabla de resultados

Bibliografía

[1] Repositorio del proyecto. https://github.com/joseangeldiazg/

[2] Website de Kaggle https://www.kaggle.com

[3] Competicion en Kaggle. https://www.kaggle.com/c/intel-mobileodt-cervical-cancer-screening