

REDES DE NEURONAS ARTIFICIALES

Curso 2021-22

PRÁCTICA II.

PARTE 1. PROBLEMA DE CLASIFICACIÓN con Perceptron Multicapa Vehicle silhouettes

1. Introducción

En la primera parte de esta práctica se va a abordar un problema de clasificación clásico con Perceptron Multicapa (PM). Se trabajará con el dominio real de clasificación “Vehicle Silhouettes”. Es un dominio compuesto por 846 instancias, 18 atributos enteros y 4 clases que corresponden a distintos tipos de vehículos. El objetivo es clasificar una determinada silueta de un vehículo en uno de los cuatro tipos disponibles en este conjunto de datos: OPEL, SAAB, BUS, VAN.



A partir de imágenes de diferentes vehículos, se han extraído una serie de características que son invariantes a las rotaciones y a los cambios de escala. Estas características son los 18 atributos enteros de cada uno de los datos. En el siguiente enlace puede encontrarse una breve descripción del dominio:

<http://sci2s.ugr.es/keel/dataset.php?cod=68#sub2>

De las 846 instancias totales, 212 corresponden a la clase OPEL, 217 a la clase SAAB, 218 a BUS y 199 a VAN.

Los datos para la realización de esta parte pueden encontrarse en la sección “Práctica 2 - parte 1” en Aula Global.

Se trabajará con un “Notebook” en el entorno Colaboratory de Google (Colab).

En Aula Global se proporciona una presentación donde se explican los detalles de la metodología a seguir y del script en Python a utilizar.

2. Trabajo a realizar para la Parte I

2.1 Preparación de los datos

Antes de utilizar los datos para crear los modelos es necesario:

1. **Normalizar** los datos de entrada en el intervalo $[0,1]$ y aleatorizarlos.
2. Dividir los datos en un conjunto de entrenamiento (2/3 de los datos) y un conjunto de test (el tercio restante).
3. **Codificar la salida deseada** (la clase) de forma adecuada para representar un problema de clasificación con cuatro clases y que pueda ser procesada por el PM. Esta codificación será one-hot-encoding. Por tanto, la columna de salida con las etiquetas correspondientes a la clases se transformará en cuatro columnas binarias. **El script proporcionado realiza esta codificación.**

2.2 Clasificación con Perceptron multicapa (PM)

En esta práctica, disponemos de un conjunto de entrenamiento y uno de test. Con los datos de entrenamiento deberemos elegir el mejor modelo posible probando con diferentes combinaciones de hiperparámetros. El mejor modelo será evaluado con el conjunto de test, obteniendo los resultados finales de la práctica.

Utilizaremos PM básicos:

- Función de activación '**sigmoid**' en las capas ocultas. Se utilizará 1 o varias capas ocultas (no más de 3).
- Capa de salida con tantas neuronas como clases con función de activación '**softmax**'. Esta función es similar a la sigmoide pero las salidas están normalizadas de forma que su suma es 1. Así puede asociarse cada salida a una probabilidad asociada a la clase correspondiente. Se asignará a la instancia la clase que corresponda a la probabilidad más alta.
- Como optimizador se utilizará el descenso del gradiente estocástico '**SGD**' (actualización de pesos con cada instancia) de forma que se minimice el Error cuadrático medio (MSE).

2.2.1 Metodología

A continuación, se explica con más detalle la **metodología a seguir**:

- Se selecciona un conjunto de hiperparámetros. Para evitar una excesiva dispersión de los experimentos, solo se variarán los siguientes:
 - Razón de aprendizaje (learning rate)
 - Arquitectura de la red: número de capas y neuronas en cada capa
 - Número de ciclos (épocas)
- Se construye una red con esos hiperparámetros y se entrena con el 80% del conjunto de entrenamiento de forma que el 20% restante constituya el conjunto de validación. Esta división se realiza automáticamente al entrenar (se explica en la presentación).
- Al finalizar cada entrenamiento, se obtiene el valor de la función de error que queremos minimizar (llamada 'loss').
- Después de realizar experimentos con diferentes conjuntos de hiperparámetros, seleccionaremos los que consiguen la mínima 'loss' en el conjunto de validación.
- **Construimos el modelo final** con los mejores hiperparámetros seleccionados. Pero este modelo **será entrenado con todos los datos de entrenamiento disponibles** (incluidos los que antes se utilizaban como validación).
- **Evaluamos el modelo final** con los datos de test obteniendo como resultados:
 - Porcentaje de aciertos (accuracy)
 - Matriz de confusión
 - Otras métricas: Recall, Precision, etc.

3. Documentación a entregar relativa a la Parte I

Se entregará una memoria de la parte I de la práctica en formato pdf y un conjunto de ficheros que contengan los modelos construidos y los resultados obtenidos.

3.1 Normas de entrega de la memoria

La memoria deberá contener, al menos, un capítulo de introducción, un capítulo explicando brevemente cómo se ha realizado el preprocesamiento de los datos y las mejoras realizadas en el script básico, un capítulo para la experimentación realizada con PM donde se describa esta experimentación, así como los resultados obtenidos y su análisis. Finalmente, un capítulo con las conclusiones generales.

La memoria deberá tener un máximo de 8 páginas con formato standard.

Los resultados que deben mostrarse en la memoria son los siguientes:

- *Evolución a lo largo del aprendizaje de los errores (loss) y porcentajes de aciertos (accuracy), de entrenamiento y validación, para los experimentos más significativos.*
 - *Valores de accuracy y loss de entrenamiento y validación al finalizar el aprendizaje de todos los experimentos realizados. Esta información se puede proporcionar en forma de tabla, que resuma todos los experimentos realizados, indicando también los hiperparámetros utilizados (learning rate, núm de neuronas ocultas y capas y ciclos utilizados).*
 - *Resultados del modelo final: Accuracy y Matriz de confusión para los datos de test.*
-
- Los ficheros a entregar son los siguientes:
 - Ficheros de datos de entrenamiento y test utilizados.
 - Un fichero de texto con el script utilizado.
 - El modelo final (solo los pesos, fichero 'pesos.h5').
 - Un fichero de texto conteniendo las predicciones para el fichero de test (con la clase) y las salidas esperadas. Estas predicciones son las que realiza el modelo final.

PARTE 2. PROBLEMA DE CLASIFICACIÓN con Redes Convolucionales

El enunciado de esta parte se publicará más adelante.

La entrega de las dos partes de la práctica 2 será el día 15 de diciembre.