

## 5. Pràctica amb la xarxa MLP 1 [R]

Albert Ribes

8 de enero de 2018

Aquesta és una tasca usada com a benchmark en la literatura. Definim  $f : [-1, 1]^2 \mapsto \mathbb{R}$  com:  $f(x_1, x_2) = 4 \sin(\pi x_1) + 2 \cos(\pi x_2) + \epsilon$  on  $\epsilon \sim \mathcal{N}(0, 0.5^2)$  és soroll normal amb mitjana zero i desviació estàndar 1.

1. Entreneu una xarxa neuronal MLP amb la rutina `nnet{nnet}` per aprendre la tasca. Heu de fer 4 estudis separats, prenent conjunts d'aprenentatge de mida creixent: 100, 200, 500 i 1000, mostrejats de manera aleatòria uniformement en  $[-1, 1]^2$ . Caldrà que estimeu la millor arquitectura, cosa que podeu fer per *cross-validation*, usant regularització.

```
trc <- trainControl (
  method="repeatedcv",
  number=10,
  repeats=5)
model1 <- train (
  target ~.,
  data = df1,
  linout = TRUE,
  method='nnet',
  metric = "RMSE",
  trControl=trc)
```

Se han generado 4 dataframes con distintos tamaños (df1, df2, df3, df4) y se ha usado caret para encontrar la arquitectura más adecuada para cada uno de ellos. Haciendo 5 veces 10-fold-cross-validation se han generado 4 modelos distintos, uno para cada dataset de *training*

2. Reporteu els resultats de predicció dels 4 estudis en un conjunt de test de mida 1024 obtingut de crear exemples a intervals regulars en  $[-1, 1]^2$ .

Los resultados están en la tabla 1.

Es muy curioso que no parece haber mejoría a medida que se incrementa el tamaño de los datos de training. De hecho, los resultados parecen oscilar. Esto puede deberse al factor de azar que hay en la construcción de las redes.

El proceso de cross-validation ha resultado en las arquitecturas de la tabla 2

3. Repetiu els experiments usant regressió lineal amb i sense regularització en els mateixos conjunts de dades i compareu els resultats obtinguts amb

Cuadro 1: Resultados de testing

	RMSE	Rsquared	MAE
r1.nnet	0,268	0,993	0,214
r2.nnet	0,552	0,970	0,428
r3.nnet	0,263	0,993	0,211
r4.nnet	1,444	0,797	1,278
r1.linear	2,366	0,491	1,928
r2.linear	2,306	0,489	1,885
r3.linear	2,300	0,492	1,883
r4.linear	2,284	0,492	1,875

Cuadro 2: Arquitectura de las redes

	size	decay
model1.nnet	5	0
model2.nnet	5	0,100
model3.nnet	5	0,100
model4.nnet	5	0,0001

els de la xarxa MLP; noteu que podeu usar simplement la rutina `nnet` amb `size=0`.

Los resultados están en la tabla 1.

En el caso de modelos lineales con regularización sí que se mejoran a medida que los datos de training se incrementan. Con esto se ve que los modelos lineales son más estables que las redes neuronales.

De todos modos, las redes neuronales siempre tienen unos resultados mejores que los modelos lineales. Esto puede ser un indicador de que el modelo lineal no tiene suficiente potencia para representar estos datos. De hecho, los modelos que se han creado no generan nuevos datos de entrada, y no tiene suficiente flexibilidad para representar las funciones de seno y coseno.

No he sido capaz de hacer un modelo lineal con R que no regularice.