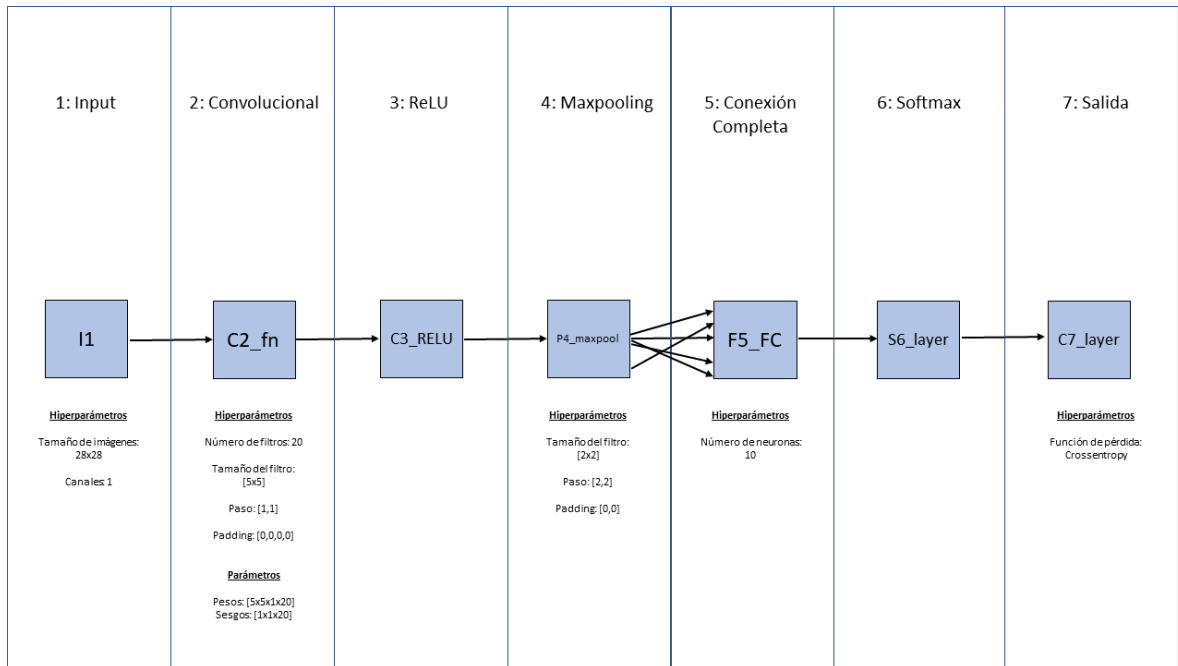


Práctica de Redes de Convolución

Juliana Quirós, Alberto

1.



2.1.

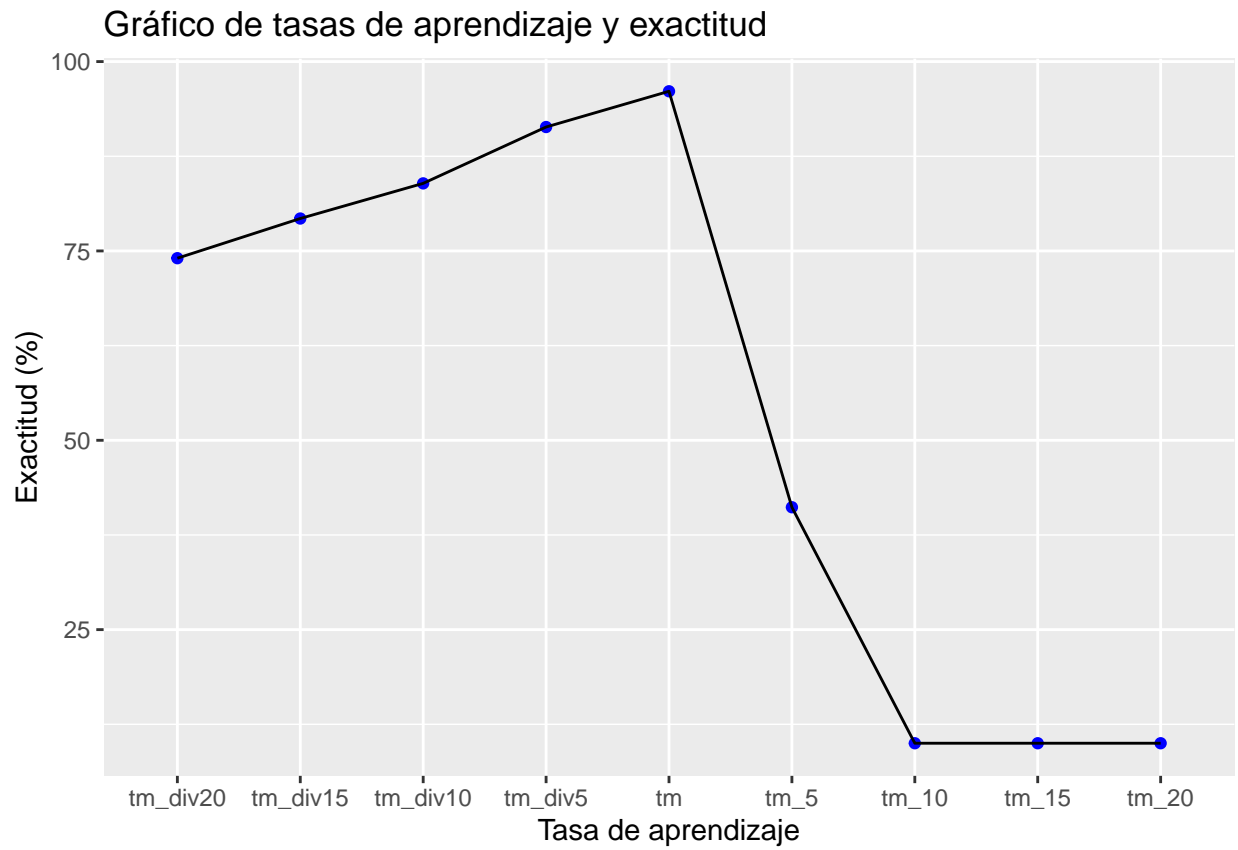
Estas fueron las tasas probadas:

tasa	exactitud
1e-05	36.76
1e-04	83.88
1e-03	96.08
1e-02	10.00
1e-01	10.00

La em es de 96.08%, con una tm=0.001.

nombre	tasa	exactitud
tm_div20	5.0e-05	74.04
tm_div15	6.7e-05	79.28
tm_div10	1.0e-04	83.92

nombre	tasa	exactitud
tm_div5	2.0e-04	91.36
tm	1.0e-03	96.08
tm_5	5.0e-03	41.16
tm_10	1.0e-02	10.00
tm_15	1.5e-02	10.00
tm_20	2.0e-02	10.00

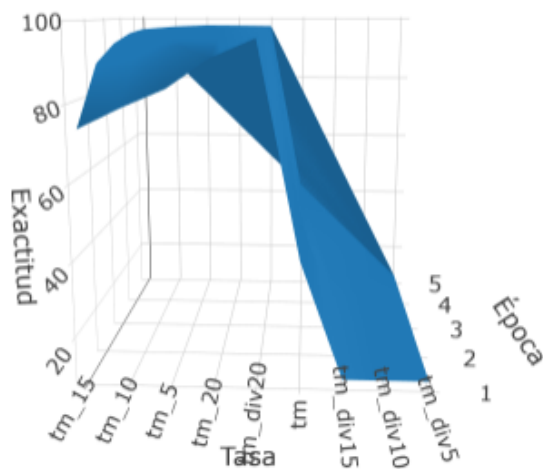


2.2

Tabla en formato long de exactitud por épocas a diferentes tasas:

nombre	tasa	época	exactitud
tm_15	5.0e-05	1	74.04
tm_10	6.7e-05	1	79.28
tm_5	1.0e-04	1	83.92
tm_20	2.0e-04	1	91.36
tm_div20	1.0e-03	1	96.08
tm	5.0e-03	1	41.16
tm_div15	1.0e-02	1	10.00
tm_div10	1.5e-02	1	10.00

nombre	tasa	época	exactitud
tm_div5	2.0e-02	1	10.00
tm_15	5.0e-05	2	88.72
tm_10	6.7e-05	2	90.20
tm_5	1.0e-04	2	93.68
tm_20	2.0e-04	2	96.68
tm_div20	1.0e-03	2	97.08
tm	5.0e-03	2	43.92
tm_div15	1.0e-02	2	10.00
tm_div10	1.5e-02	2	10.00
tm_div5	2.0e-02	2	10.00
tm_15	5.0e-05	3	93.20
tm_10	6.7e-05	3	94.04
tm_5	1.0e-04	3	96.32
tm_20	2.0e-04	3	97.24
tm_div20	1.0e-03	3	97.52
tm	5.0e-03	3	39.32
tm_div15	1.0e-02	3	10.00
tm_div10	1.5e-02	3	10.00
tm_div5	2.0e-02	3	10.00
tm_15	5.0e-05	4	95.48
tm_10	6.7e-05	4	96.24
tm_5	1.0e-04	4	97.28
tm_20	2.0e-04	4	97.48
tm_div20	1.0e-03	4	97.40
tm	5.0e-03	4	24.36
tm_div15	1.0e-02	4	10.00
tm_div10	1.5e-02	4	10.00
tm_div5	2.0e-02	4	10.00
tm_15	5.0e-05	5	96.36
tm_10	6.7e-05	5	97.28
tm_5	1.0e-04	5	97.72
tm_20	2.0e-04	5	97.64
tm_div20	1.0e-03	5	97.48
tm	5.0e-03	5	40.12
tm_div15	1.0e-02	5	10.00
tm_div10	1.5e-02	5	10.00
tm_div5	2.0e-02	5	10.00



Medias de exactitud por épocas:

exactitud_10	exactitud_20	exactitud_30	exactitud_40	exactitud_50
55.09333	60.03111	60.84889	59.80444	61.84444

2.3

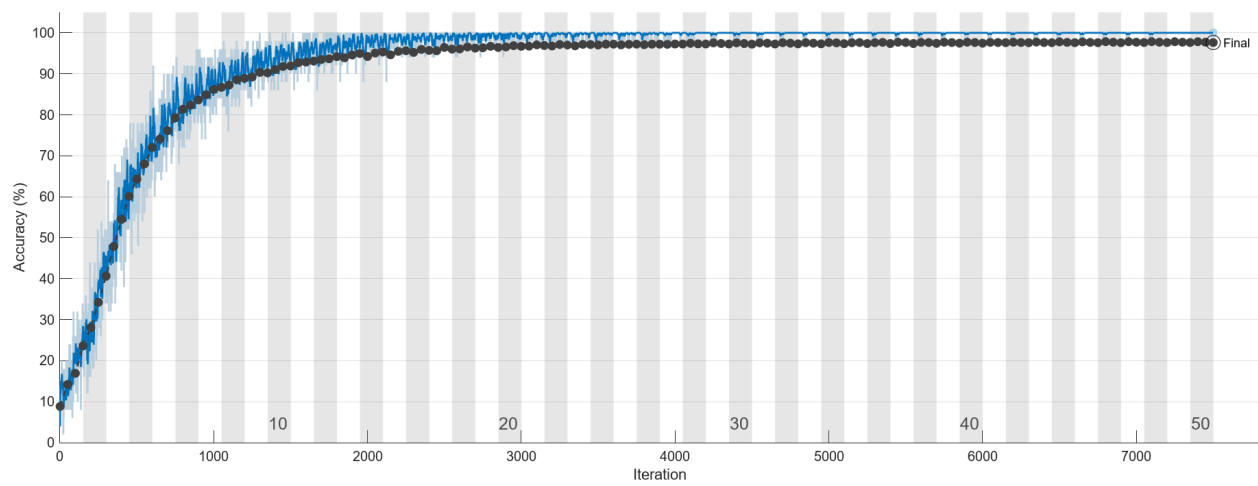
Como podemos observar, obtenemos una mayor exactitud con tasas de aprendizaje más pequeñas que con múltiplos de la máxima. Esto puede deberse a que el algoritmo converge en una solución subóptima (mínimo local) en los primeros casos, frente a la omisión directa de dichos mínimos en los segundos casos, que inducen errores de detección y clasificación.

Observamos a su vez, que la tasa que da lugar a la exactitud máxima se mantiene hasta llegar a 30 épocas. A partir de las 40, $tm/5$ supera ligeramente a tm . Este fenómeno se debe a que el algoritmo ha dispuesto de mayor tiempo de entrenamiento.

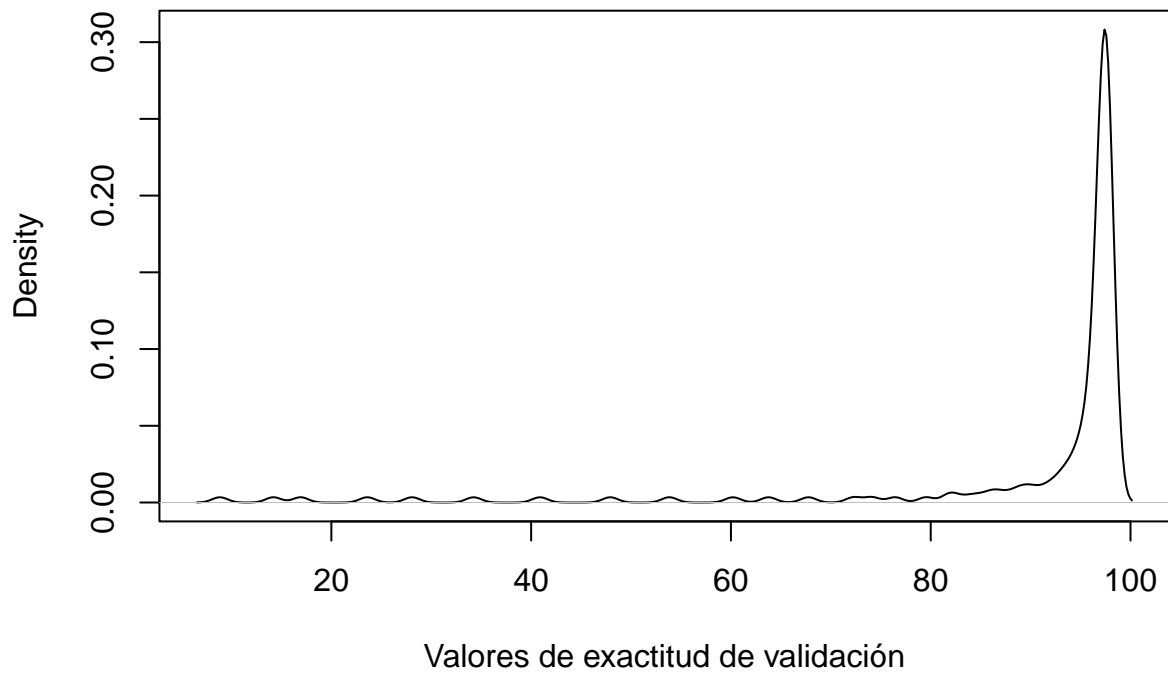
La media de exactitud más alta (61.8444) se produce con 50 iteraciones, lo cual confirma lo anteriormente expuesto.

3.1

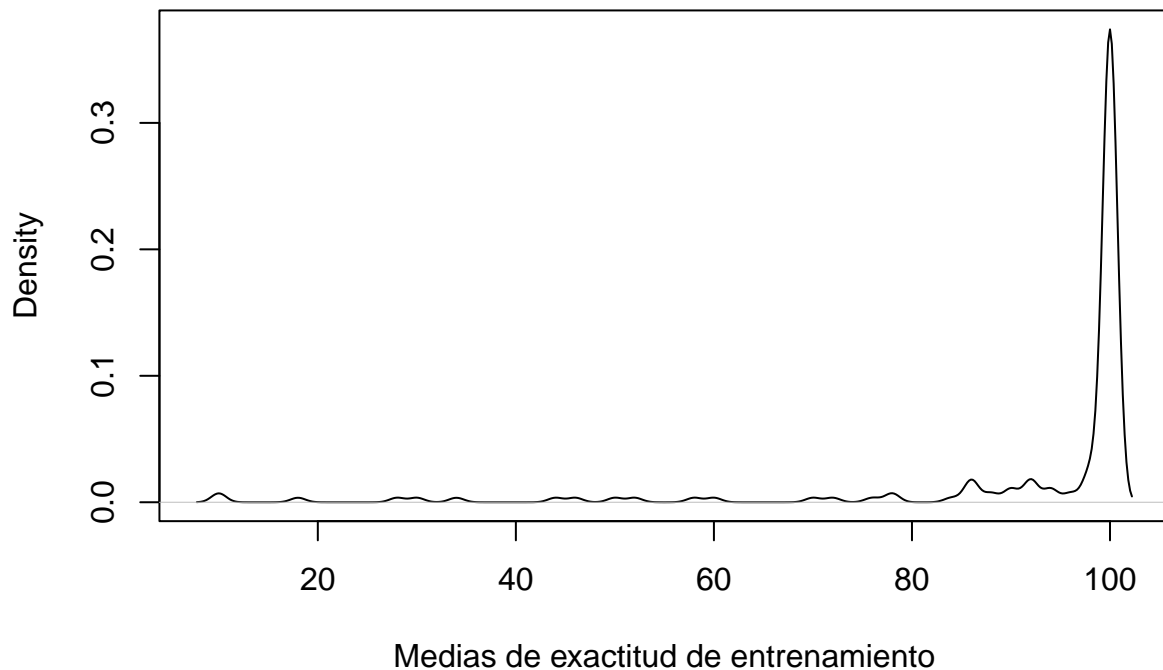
Basándome en el ejercicio anterior, empleo $tm/5$ (0.0002) y 50 épocas:



3.2



```
plot(density(med_train), main = "", xlab = "Medias de exactitud de entrenamiento")
```



Empleo la prueba no paramétrica de los signos para 2 muestras, dado que ambas son muy asimétricas.

```
wilcox.test(med_train, val_acc, alternative = "two.sided", exact = FALSE,
            correct = FALSE)
```

```
##
## Wilcoxon rank sum test
##
## data: med_train and val_acc
## W = 17805, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Por lo tanto, con un 95% de confianza rechazo la H_0 de igualdad de medianas, por lo que existe evidencia estadística de sobreajuste.

```
minilotes <- seq(1, 151)
```

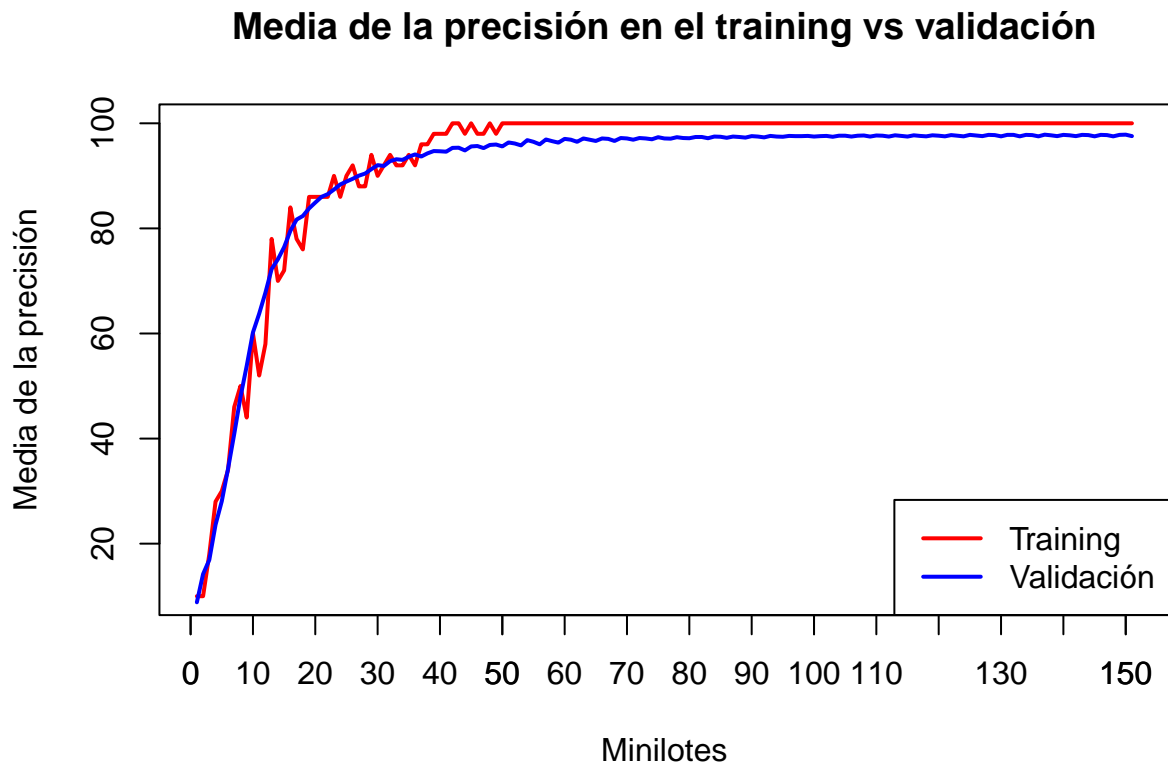
```
plot(minilotes, med_train, type = "l", col = "red", lwd = 2,
     xlab = "Minilotes", ylab = "Media de la precisión", main = "Media de la precisión en el training vs
```

```
lines(minilotes, val_acc, type = "l", col = "blue", lwd = 2)
```

```

legend("bottomright", legend = c("Training", "Validación"),
      col = c("red", "blue"), lwd = 2)
axis(1, at = seq(0, 151, by = 10))

```



```

df_trainval <- cbind(minilotes, med_train, val_acc)
df_trainval <- df_trainval[1:50, ]
knitr::kable(df_trainval, format = "markdown", digits = 6, row.names = FALSE,
              align = c("l", "c", "r"))

```

minilotes	med_train	val_acc
1	10	8.84
2	10	14.20
3	18	16.92
4	28	23.60
5	30	28.08
6	34	34.24
7	46	40.88
8	50	47.92
9	44	53.84
10	60	60.20
11	52	63.76
12	58	67.76
13	78	72.28

minilotes	med_train	val_acc
14	70	74.12
15	72	76.44
16	84	79.52
17	78	81.68
18	76	82.36
19	86	83.80
20	86	84.92
21	86	86.00
22	86	86.52
23	90	87.40
24	86	88.36
25	90	88.92
26	92	89.44
27	88	90.04
28	88	90.44
29	94	91.28
30	90	92.04
31	92	91.88
32	94	92.80
33	92	93.16
34	92	93.04
35	94	93.64
36	92	94.08
37	96	93.68
38	96	94.28
39	98	94.72
40	98	94.68
41	98	94.60
42	100	95.32
43	100	95.36
44	98	94.84
45	100	95.60
46	98	95.68
47	98	95.28
48	100	95.88
49	98	95.96
50	100	95.60

Podemos observar como a partir de, aproximadamente, el minilote 36 (iteración 5400), comienzan a separarse ambos conjuntos de forma sistemática.