

SELECCIÓN DE OPTIMIZADORES PARA LA RED SOCO Y CONJUNTO DE DATOS ISIC

Por Antonio Jesús García Nieto

[illegible]

TABLA DE CONTENIDO

Control de versiones.....	2
Introducción	4
Planteamiento del estudio	4
Resultados	5
Conclusión del experimento.....	6
Bibliografía.....	6

INTRODUCCIÓN

En este experimento estudiaremos la variación de la precisión en función de la selección de optimizadores para nuestro problema en concreto, es decir, el conjunto de datos ISIC y la tarea de clasificación múltiple. Nos basaremos en (Jayalakshmi & Kumar, 2019). Este estudio, entre otras cosas, realiza una exploración en lo referente a optimizadores y ratios de aprendizaje para el problema específico de la detección y clasificación de lesiones de piel.

La hipótesis que pretendemos validar en este estudio es:

1. *Existe un optimizador que tiene un mejor desempeño que RMSProp para nuestro problema.*

PLANTEAMIENTO DEL ESTUDIO

El estudio consistirá en una comparación de la precisión media entre la red SOCO, configurada según los experimento 1, con diversos optimizadores.

Se seguirá el mismo esquema de entrenamiento que en el experimento 1. Recordemos que el este es:

- El conjunto de entrenamiento = source (60% del total del conjunto de datos),
- El conjunto de reentrenamiento = train (70% del 40% restante del conjunto de datos),
- El conjunto de test = test (21% del 40% restante del conjunto de datos)
- El conjunto de validación = validación (9% del 40% restante del conjunto de datos)

Los hiperparámetros serán los siguiente:

- Ratio de aprendizaje de 0.0001, para todos los optimizadores.
- Épocas: 15 para entrenamiento y para transferencia.
- Tamaño de lote: 16
- Dropout 0.4

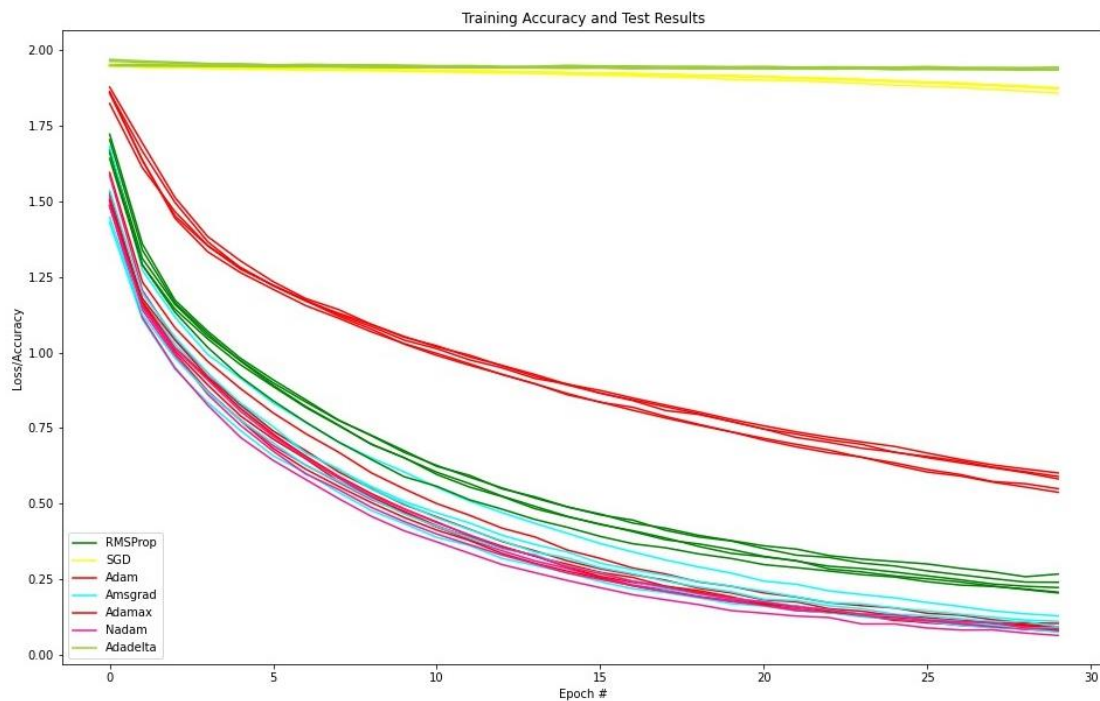
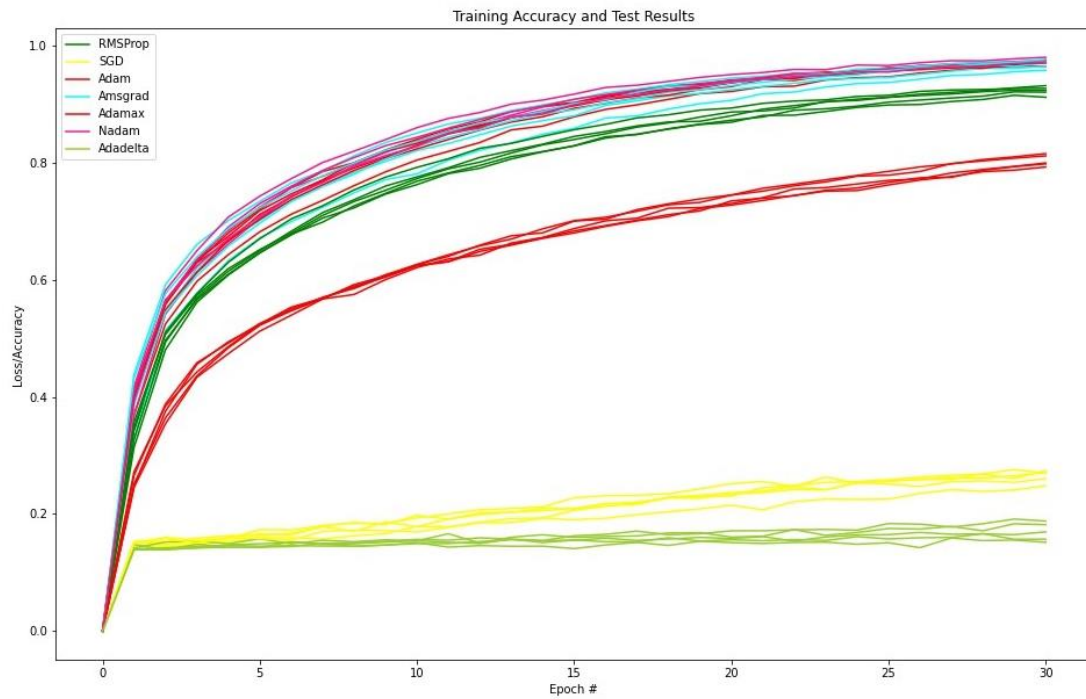
Los optimizadores que se van a probar son:

- RMSProp
- SGD
- Adam
- Amsgrad
- Adamax
- Nadam
- Adadelta

Para asegurar la robustez de la prueba se entrenarán cinco modelos con cada optimizador.

RESULTADOS

En primer lugar, se comparará la precisión sobre los datos de entrenamiento y el valor de pérdida de todos los modelos entrenados.



Como podemos apreciar, algunos optimizadores como Adamax, Adadelta o SGD son muy ineficientes al menos en el corto plazo.

Los resultados globales al evaluar sobre el conjunto de datos de test son los siguientes:

	RMSProp	SGD	Adam	Amsgrad	Adamax	Nadam	Adadelata
0	0.900262	0.300379	0.923885	0.916594	0.790609	0.923301	0.186060
1	0.888014	0.307961	0.927384	0.928259	0.790026	0.930009	0.196850
2	0.905220	0.290172	0.931467	0.929134	0.784485	0.928259	0.205599
3	0.903470	0.284048	0.924176	0.927967	0.805774	0.926509	0.192184
4	0.905220	0.299504	0.924759	0.933800	0.787985	0.927384	0.227472

Por otro lado, la precisión media de cada optimizador es la siguiente:

```
RMSProp    0.900437
SGD         0.296413
Adam        0.926334
Amsgrad     0.927151
Adamax      0.791776
Nadam       0.927092
Adadelata   0.201633
dtype: float64
```

Podemos ver que el top 4 optimizadores es Amsgrad, Nadam, Adam, RMSProp por orden de precisión.

CONCLUSIÓN DEL EXPERIMENTO

Los resultados obtenidos han sido muy semejantes a los del estudio (Jayalakshmi & Kumar, 2019). Adam ha tomado un buen desempeño, y en ambos casos superior a RMSProp.

El rendimiento se incrementa en un 2.6% cuando se usa Amsgrad frente a RMSProp. Por lo tanto, podemos afirmar que la hipótesis enunciada se cumple, y que en lugar de usar RMSProp podemos pasar a utilizar Amsgrad como optimizador.

No obstante, este estudio nos deja otra pequeña conclusión en el aire, y es que 30 épocas de entrenamiento (15 mediante scratching y 15 mediante transfer learning) no son suficientes para conseguir el mejor valor en cuanto a precisión media. Una idea clave para siguientes estudios puede ser trabajar con un rango variable de épocas mediante planificación del entrenamiento o paradas preestablecidas en el algoritmo de optimización.

BIBLIOGRAFÍA

Jayalakshmi, G. S., & Kumar, V. S. (2019, February 1). Performance analysis of convolutional neural network (CNN) based cancerous skin lesion detection system. *ICCIDS 2019 - 2nd International Conference on Computational Intelligence in Data Science, Proceedings*.
<https://doi.org/10.1109/ICCIDS.2019.8862143>