



Vision por Computadora II

Modelo de clasificación
de frutas

Juan Pablo Alianak

01

...

02

...

03

...

04

...

05

...

Selección del Dataset

Importación y generación de los datos

Exploración del Dataset

Cantidad de imágenes en train, valid y test. Balance de clases

Entrenamiento de modelos

ResNet18, ResNet50 e Inception V3 con y sin Data Aumentación

Transfer Learning

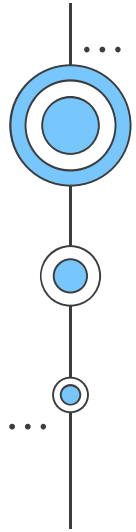
Con el modelo de mejor performance

Optimización de Hiperparametros

You could describe the topic of the section here

Contenido del TP





01

Selección del Dataset

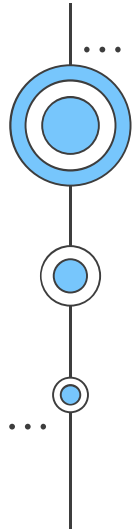




Fruit Dataset

El dataset fue seleccionado de la base de datos de Roboflow. Contiene 3155 imagenes de 11 clases distintas de frutas (bananas, manzanas, mango, naranja, etc).

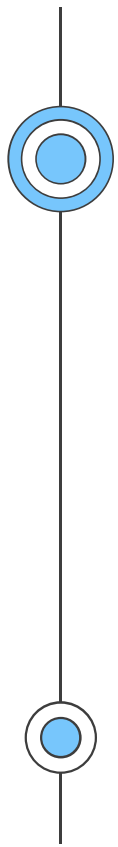
...



02

Exploracion del Dataset





Dataset

Total

100%
3155 imagenes



Train

70%
2200 imagenes

...



Valid

20%
663 imagenes

...

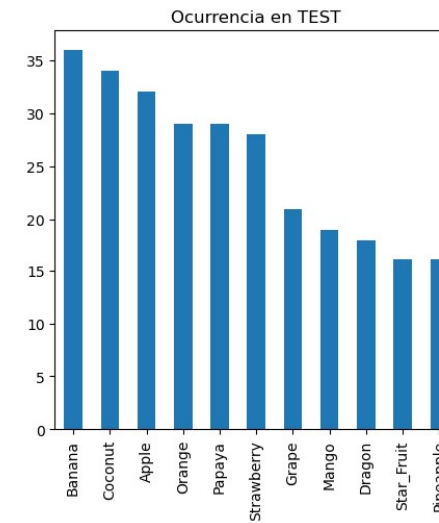
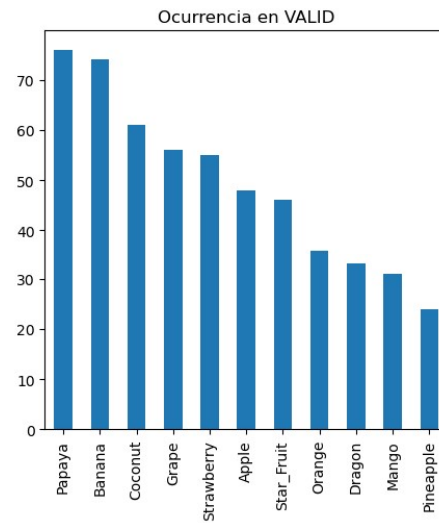
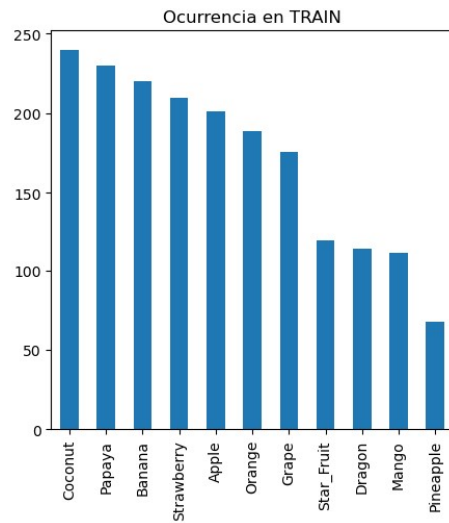


Test

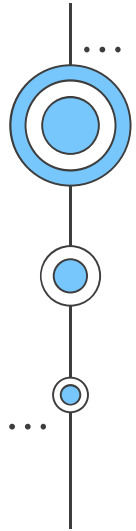
10%
316 imagenes

...

Histograma de ocurrencia de clase en TRAIN – VALID – TEST



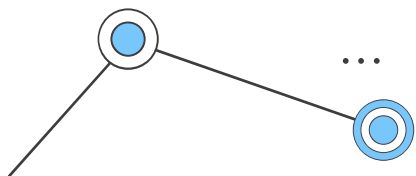
El dataset presenta un balance aceptable para cada una de las clases, con lo cual no es necesario hacer un balance de las mismas.



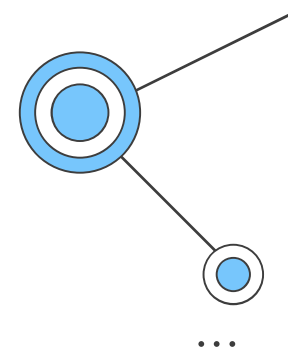
03

Entrenamiento de modelos





Modelos entrenados



01

ResNet18

Red neuronal convolucional con 18 capas de profundidad.
Aprox. 11 millones de parámetros.

02

ResNet50

Red neuronal convolucional con 50 capas de profundidad.
Aprox. 23 millones de parámetros.

03

Inceptionv3

Red neuronal convolucional con 48 capas de profundidad.
Aprox. 24 millones de parámetros.

Metricas – Accuracy

Todos los modelos fueron entrenados con los siguientes
parametros:

➤ Epoch = 20 - Optimizador = Adams - Lr = 0.001 - Metrica = Accuracy

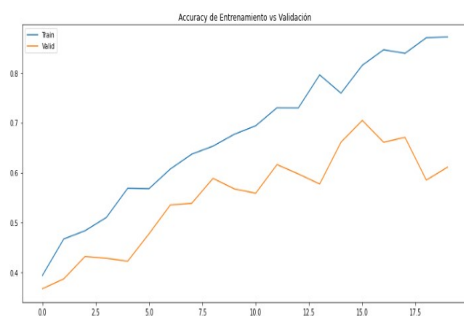
	ResNet18	ResNet50	Inception v3
Sin DA – Train	0.87	0.88	0.85
Sin DA – Valid	0.67	0.50	0.67
Sin DA – Test	0.51	0.47	0.70
Con DA – Train	0.67	0.62	0.73
Con DA – Valid	0.56	0.54	0.65
Con DA – Test	0.60	0.59	0.64

Parámetros utilizados para realizar Data Aumentation

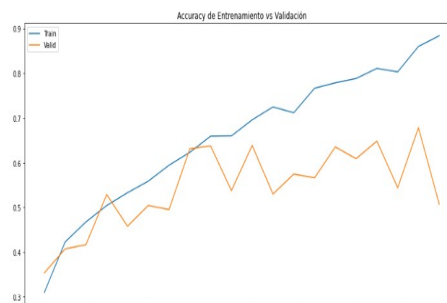
➤ RandomHorizontalFlip – RandomResizedCrop – ColorJitter - Normalize

Metricas – Accuracy

ResNet 18



ResNet 50

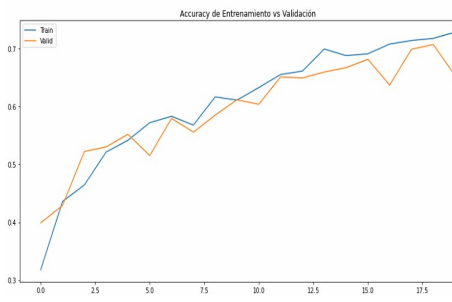
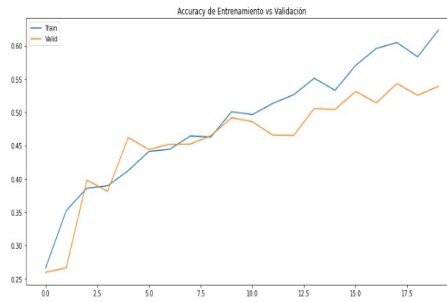
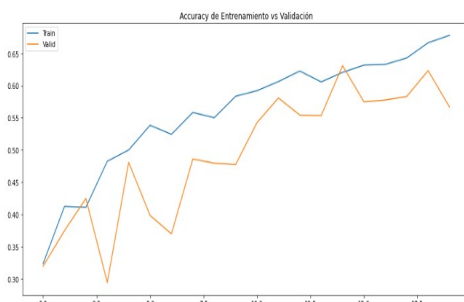


Inception V3

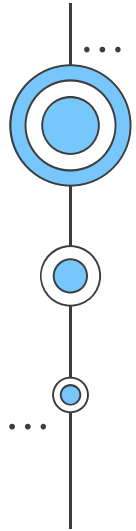


Sin DA

Con DA



Se observa en todos los modelos una clara tendencia al sobre-entrenamiento en los casos Sin DA, tendencia que se corrige cuando aplicamos DA (mejor generalización del modelo). Otro punto a destacar es que las métricas son algo menores con DA, deberíamos correr el modelo por mas epoch

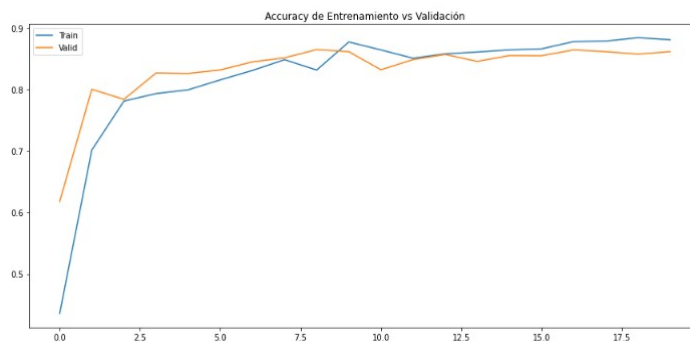
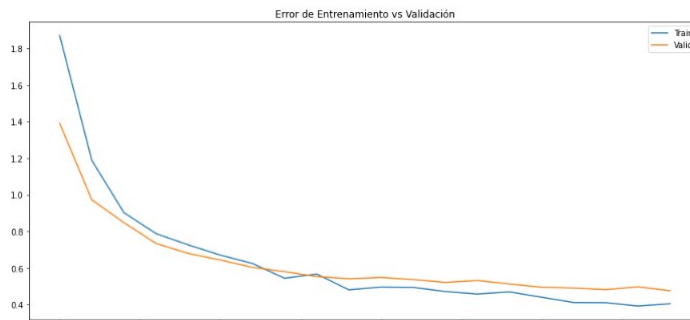


04

Trasfer Learning



Transfer Learning



Pre-training weights = IMAGENET1K_V1
Last layer = nn.Linear (last_layer_in_features, 11)
Estrategia = "Feature extraction"

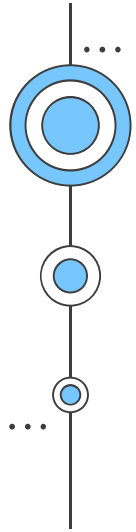
Inception v3

Train 0.88

Valid 0.86

Test 0.86

Es notable en el caso de hacer Transfer Learning los resultados del modelo, incluso estos seguramente mejorarían si corriésemos el modelo por mas epoch. Vemos una coincidencia plena entre train, valid y test concluyendo que nuestro modelo generaliza bien.



06

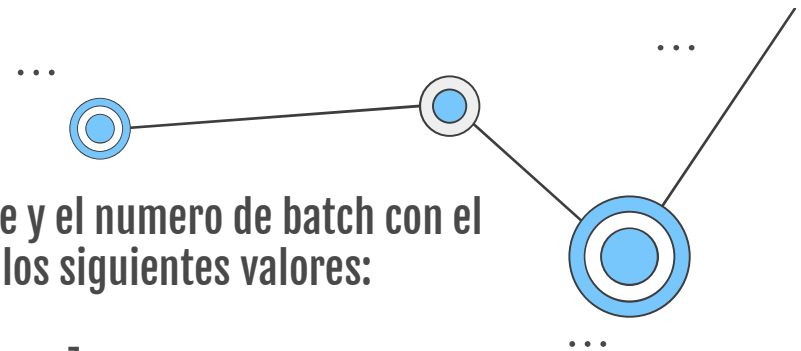
Optimizacion de hiperparametros



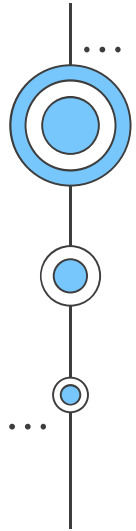
La optimización de parámetros se hizo sobre el Learnign rate y el numero de batch con el mismo modelo de Inception V3 utilizado anteriormente con los siguientes valores:

$Lr = [1e-1, 1e-4]$ - Batch = [16, 32, 64]

Debido a los bajos recursos computacionales de Google Colab, las pruebas de optimización se hicieron sobre un numero bajo de epoch, con lo cual los resultados de Loss y Accuracy logrados fueron inferiores a logrados con Transfer Learning. Seguramente dejando al modelo que corra por mas epochs y mas num-samples los resultados podrían ser muy superiores.



Inception v3	
Train	0.68
Test	0.51
Lr Optimo = 0.0006368 Batch Optimo = 32	



07

Conclusiones



- A lo largo del TP, implementamos muchas de las herramientas desarrolladas dentro del bimestre consiguiendo resultados satisfactorios.
- Se obtuvo un modelo que tiene métricas aceptables para la cantidad de épocas de entrenamiento.
- Algo a destacar a lo largo del TP, fue la falta de recursos computacionales en Google Colab, haciendo muy lenta cada una de las pruebas. Quizás, con una configuración diferente de los recursos, hubiese sido mas rápido el avance pudiendo implementar y contrastar mas modelos.
- Queda a futuro hacer el deployment del modelo para que pueda ser consumido desde diferentes ubicaciones.

