

*MANUAL
CLASIFICACIÓN DE
IMÁGENES USANDO
NVIDIA DIGITS*

Jorge Alvarado Díaz

Versión: 1.1 - 02-12-2020

Tabla de contenido

<i>Creación del Dataset de imágenes</i>	2
Preparación del Dataset de Imágenes	2
Transferir imágenes a la máquina DIGITS	2
Para conectarse por primera vez a la máquina	3
Transferir archivos a la máquina	4
Crear Dataset en DIGITS	6
<i>Creación del Modelo de Clasificación</i>	8
<i>Probar Modelo</i>	12
Probar una única imagen	12
Probar un conjunto de imágenes	13
<i>Personalizar Red Neuronal en DIGITS</i>	15
<i>Solución de Problemas / Preguntas Frecuentes</i>	17
Filezilla no es capaz de conectarse con la máquina DIGITS	17
No soy capaz de acceder a DIGITS desde el navegador	17
Error “Model diverged with train/model/loss/train/model/loss/loss = nan : Try decreasing your learning rate” al entrenar el modelo con DIGITS TensorFlow	17

Creación del Dataset de imágenes

Preparación del Dataset de Imágenes

En caso de no tener un dataset de imágenes previamente creado, deberemos preparar uno. Para crearlo, deberemos seguir los siguientes pasos:

1. En primer lugar, deberemos crear una carpeta en nuestro dónde irán las imágenes
2. A continuación, dentro de esta carpeta, deberemos crear subdirectorios por cada clase que vamos a clasificar
3. Dentro de cada subdirectorio, copiaremos las imágenes correspondientes cada clase. Por ejemplo, tenemos tres clases A, B y C, entonces las imágenes de la clase A irán en el subdirectorio A, las de la B en el subdirectorio B, etc.

Notas Importantes:

- Todas las imágenes deberán tener la misma resolución
- El formato de archivo de imagen deberá ser PNG o JPEG
- Es recomendable que las imágenes tengan el tamaño de entrada que acepte la red neuronal a utilizar. Por ejemplo: *AlexNet* acepta imágenes de 256 x 256 píxeles y *GoogLeNet* acepta imágenes de 224 x 224 píxeles.

Transferir imágenes a la máquina DIGITS

Una vez que tengamos preparado el directorio con las imágenes, deberemos transferirlo a la máquina dónde se está ejecutando DIGITS.

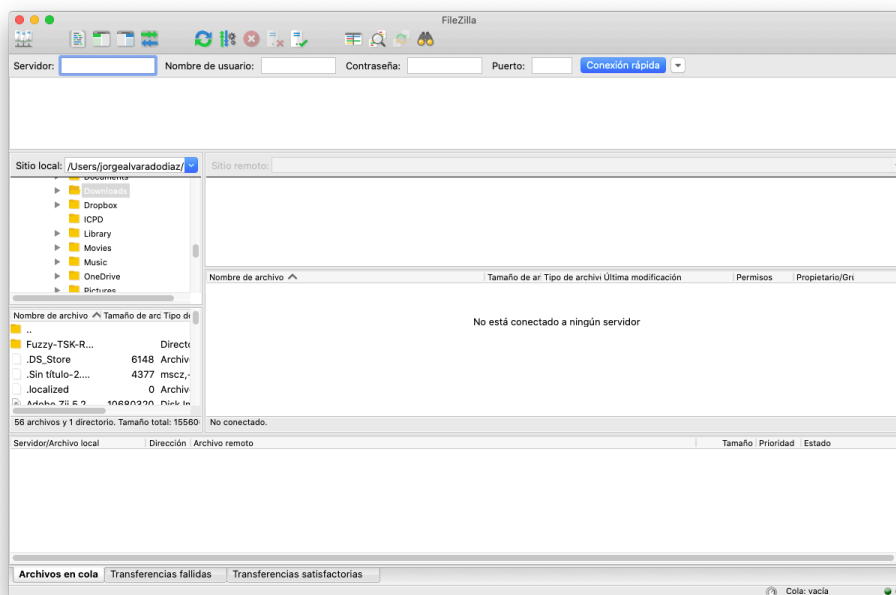
Existen numerosas herramientas para transferir ficheros mediante SCP/SFTP, entre las cuales se encuentran:

- Filezilla Client (Windows/Mac/Linux): <https://filezilla-project.org/>
- WinSCP (Windows): <https://winscp.net/eng/docs/lang:es>

En este manual haremos uso de FileZilla al tratarse de una herramienta multiplataforma y de fácil uso.

Nos descargamos el programa desde su página web <https://filezilla-project.org/download.php?type=client> y lo instalamos en nuestro ordenador.

Una vez instalado, lo ejecutamos y nos deberá aparecer la siguiente pantalla:

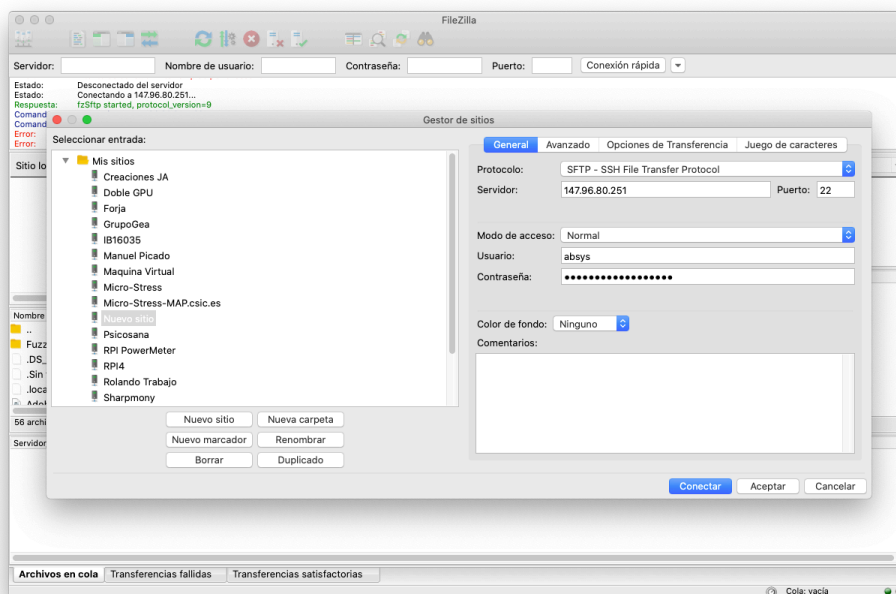


A continuación, nos vamos al menú y seleccionamos *Archivo -> Gestor de Sitios*.

Si usted ya se ha conectado previamente a la máquina, salte al apartado “Transferir archivos a la Máquina”.

Para conectarse por primera vez a la máquina

En la ventana que se nos abre, pulsamos en el botón “Nuevo Sitio”.



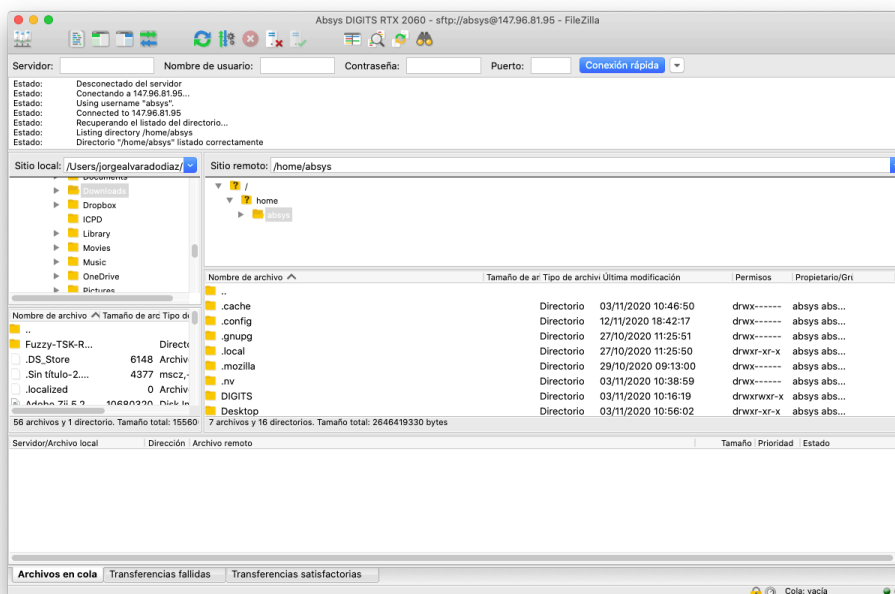
Aquí deberemos proporcionar los datos de acceso de la máquina que esta ejecutando DIGITS:

- **Protocolo:** *SFTP – SSH File Transfer Protocol*
- **Servidor:** dirección IP de la máquina
- **Puerto:** 22
- **Usuario:** su nombre de usuario en la máquina
- **Contraseña:** la contraseña del su usuario en la máquina

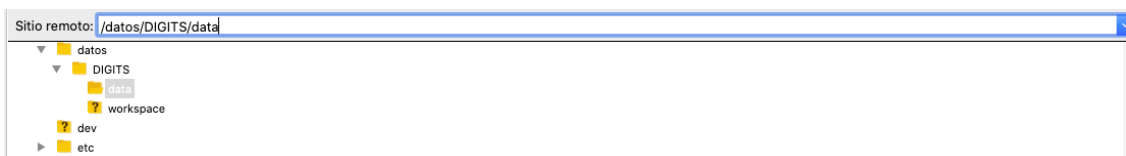
Para terminar, le asignamos un nombre a la conexión pulsando en el botón “Renombrar” y terminamos pulsando en “Aceptar”.

Transferir archivos a la máquina

Nos vamos de nuevo al gestor de sitios de Filezilla y seleccionamos la conexión a la máquina previamente creada y pulsamos en el botón “Conectar”. Si todo ha ido bien, nos deberá aparecer listado de archivo de nuestro directorio en la máquina:



A continuación, tecleamos en donde pone “Sitio Remoto” la ruta donde se almacenan los dataset de DIGITS que es /datos/DIGITS/data:



Ahora para transferir las imágenes, simplemente arrastramos la carpeta desde nuestro ordenador hacia ese directorio tal como se muestra en la siguiente imagen:



Nota: tenga cuidado de no arrastrar la carpeta dentro del directorio de otro dataset, esto podría causar problemas de funcionamiento en DIGITS.

Nota Importante: dependiendo del número de imágenes y de su conexión a internet, es posible que este proceso pueda demorar bastante tiempo en completarse. Una vez que la transferencia finalice, no deberá quedar ningún archivo en la cola de transferencia.

Crear Dataset en DIGITS

Ahora, nos conectamos introduciendo la dirección de la máquina DIGITS que nos ha sido proporcionada usando cualquier navegador web.

En la pantalla principal de DIGITS, seleccionamos la pestaña “Datasets” y pulsamos en el botón *New Dataset -> Images -> Classification* tal como se muestra en la siguiente imagen:



Es posible que DIGITS nos pida a continuación un nombre de usuario, nos inventamos uno y pulsamos en “Submit”. **Nota:** es recomendable que cada vez que accedamos a DIGITS, usemos el mismo nombre de usuario.

En la siguiente pantalla se nos muestran las diferentes opciones para la creación del dataset:

New Image Classification Dataset

The screenshot shows the 'New Image Classification Dataset' configuration form. It is divided into several sections:

- Image Type:** A dropdown menu set to 'Color'.
- Image size (Width x Height):** Two input fields, both set to '256'.
- Resize Transformation:** A dropdown menu set to 'Squash'.
- See example:** A blue button.
- Use Image Folder / Use Text Files / Use S3:** Three radio buttons, with 'Use Image Folder' selected.
- Training Images:** A text input field containing '/data/dataset_pruebas'.
- Minimum samples per class:** An input field set to '2'.
- Maximum samples per class:** An empty input field.
- % for validation:** An input field set to '15'.
- % for testing:** An input field set to '3'.
- Separate validation images folder:** An unchecked checkbox.
- Separate test images folder:** An unchecked checkbox.
- DB backend:** A dropdown menu set to 'LMDB'.
- Image Encoding:** A dropdown menu set to 'PNG (lossless)'.
- Group Name:** A text input field containing 'Pruebas'.
- Dataset Name:** A text input field containing 'Dataset_Pruebas'.
- Create:** A blue button at the bottom.

Las opciones más importantes son:

- **Image Type:** imágenes en color o en escala de grises (por defecto en color)
 - **Image size:** tamaño de las imágenes (por defecto 256 x 256)
 - **Resize Transformation:** método de transformación de imágenes al cambiar la resolución:
 - Crop: recortar bordes
 - Squash: comprimir imagen (por defecto)
 - Fill: rellenar bordes
 - Half Fill, Half Crop: 50% fill y 50% crop
- Nota:** se puede ver que hace cada método pulsando sobre “See example”
- **Training Images:** aquí indicamos la ruta al directorio dentro de DIGITS que contiene las imágenes del dataset. Deberá ser /data/<nombre_directorio_dataset> dónde <nombre_directorio_dataset> hay que sustituirlo por el nombre que le hemos dado a la carpeta del dataset
 - **% for validation:** porcentaje de imágenes para validación (recomendado entre 10 – 25 %)
 - **% for testing:** porcentaje de imágenes para testing (recomendado entre 3 – 5 %)
 - **Image Encoding:** indicar el formato de archivo que tienen las imágenes (PNG o JPEG)
 - **Group Name:** aquí hay que asignar un nombre al grupo que pertenece el dataset
 - **Dataset Name:** nombre que le vamos a dar al dataset

Nota Importante: las opciones de *Image Type* y *Image Size* deberán ser adaptadas a la red que se va a utilizar. Por ejemplo, para las redes:

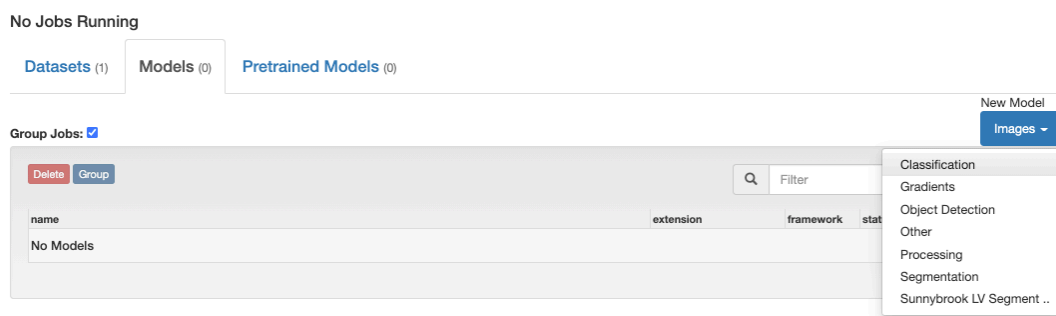
- **LeNet:** Imágenes en escala de grises y tamaño 28 x 28
- **AlexNet:** Imágenes en color y tamaño 256 x 256
- **GoogLeNet:** Imágenes en color y tamaño 224 x 224

Pulsamos en “Create” para crear el Dataset en DIGITS. Debemos esperar a que DIGITS procese las imágenes y una vez que haya terminado, deberá aparecer una pantalla dónde podremos visualizar la información del dataset recién creado:

Job Information	Parse Folder (train/val/test)
<p>Job Directory /workspace/jobs/20201119-103513-f1e6</p> <p>Image Dimensions 256x256 (Width x Height)</p> <p>Image Type Color</p> <p>Resize Transformation Squash</p> <p>DB Backend Imdb</p> <p>Image Encoding png</p> <p>DB Compression none</p> <p>Dataset size 687 MB</p>	<p>Folder /data/dataset_pruebas</p> <p>Number of Categories 3</p> <p>Training Images 5675</p> <p>Validation Images 1037 (15.0%)</p> <p>Test images 208 (3.0%)</p>

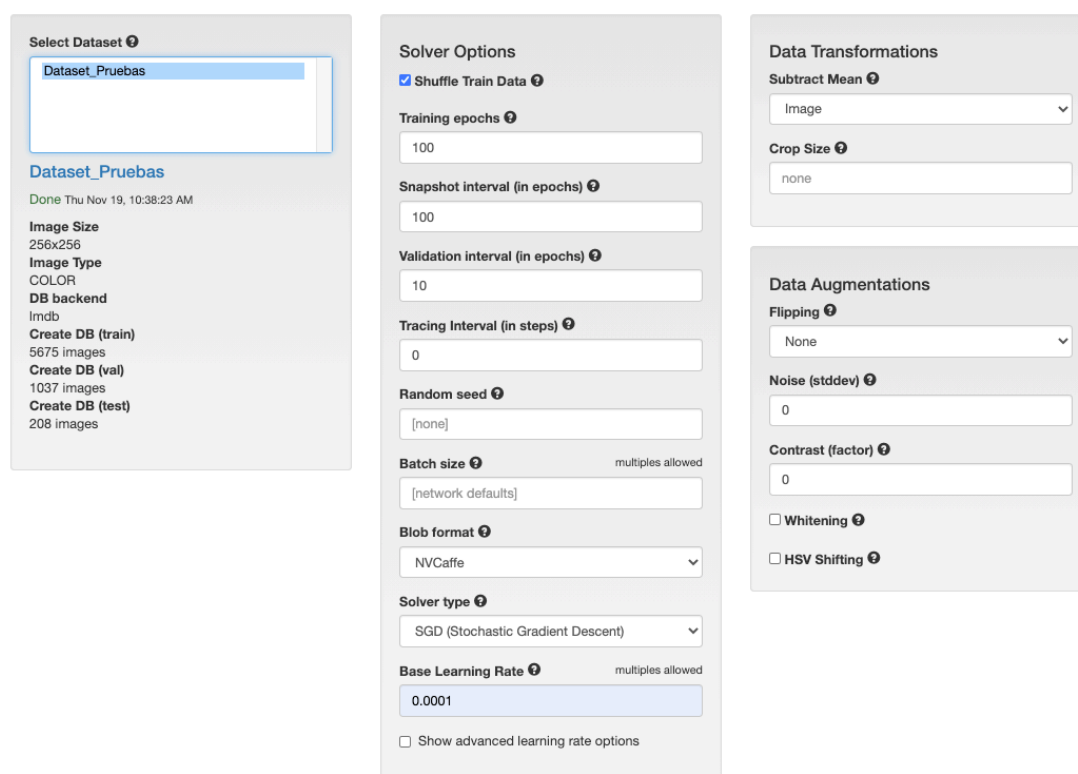
Creación del Modelo de Clasificación

Una vez creado el dataset de imágenes, nos vamos a la página principal de DIGITS donde seleccionamos en la pestaña “Models” y pulsamos sobre *New Model -> Images -> Classification* tal como se muestra en la siguiente captura de pantalla:



A continuación, se nos muestran las opciones para crear el nuevo modelo de clasificación

New Image Classification Model



Las opciones más importantes son las siguientes:

- **Select Dataset:** aquí deberemos seleccionar el dataset que vamos a emplear para entrenar el modelo
- **Solver Options:**
 - **Training epochs:** número de épocas (iteraciones) se va a entrenar la red neuronal. Para empezar, dejar en 100 e ir ajustando

- **Snapshot epochs:** intervalo de épocas para crear capturas de la red neuronal. Se recomienda establecer el mismo valor que *Training epochs* debido a que un valor más pequeño puede generar numerosos ficheros de gran tamaño, lo que puede provocar que la máquina se quede sin espacio disponible en disco
- **Validation interval:** intervalo de épocas para validación. Se recomienda que sea un 10% con respecto al valor *Training epochs*. Por ejemplo: para 100 épocas de entrenamiento establecer un intervalo de validación de 10 épocas
- **Solver type:** algoritmo de entrenamiento. Para empezar, se recomienda dejar en *Stochastic Gradient Descendent (SGD)*
- **Base Learning Rate:** tasa de aprendizaje. Se recomienda establecer un valor de 0.01 o menor (para TensorFlow se recomienda un valor de 0.001 o menor)

Data Transformations: transformación de los datos

- **Subtract Mean:** normalización de las imágenes: por imagen o píxel medios
- **Crop Size:** [opcional] tamaño en píxeles para realizar un recorte cuadrado aleatorio a las imágenes.

Data Augmentations (sólo DIGITS TensorFlow): aumento de datos creando nuevas imágenes. **Nota:** Se recomienda usar estas opciones únicamente si existe *overfitting* (sobreajuste) al entrenar previamente el modelo.

- **Flipping:** voltear imágenes para crear nuevas imágenes (se recomienda utilizar Horizontal):
 - None: no voltear
 - Horizontal: voltear horizontalmente
 - Vertical: voltear verticalmente
 - Horizontal and/or Vertical: voltear horizontal y verticalmente
- **Noise (stddev):** crear nuevas imágenes agregando ruido gaussiano blanco (Additive White Gaussian Noise - AWGN) durante el pre-procesado de las imágenes, indicando el valor de la desviación típica (se asume los pixeles de las imágenes normalizados de 0 a 1). Se recomienda establecer un valor de 0.03.
- **Contrast (factor):** crear imágenes con diferente contraste. Se indica el factor de contraste.
- **Whitening:** crear imágenes blanqueándolas mediante la resta de su media y dividiendo por su desviación típica
- **HSV Shifting:** desplazamiento aleatorio en el espacio de color HSV

Una vez establecidos los parámetros, hay que seleccionar una red para entrenar el modelo. Aquí DIGITS nos permite utilizar varios tipos de redes:

- **Standard Networks:** redes por defecto ya pre-entrenadas
- **Previous Networks:** modelos previamente entrenados
- **Pretrained Networks:** modelos pre-entrenados
- **Custom:** red personalizada. Véase [Personalizar Red Neuronal en DIGITS](#).

En nuestro caso haremos uso de una las redes por defecto, con lo que seleccionamos una de las tres que nos ofrece DIGITS:

Standard Networks Previous Networks Pretrained Networks Custom Network

Caffe

Network	Details	Intended image size
<input type="radio"/> LeNet	Original paper [1998]	28x28 (gray)
<input type="radio"/> AlexNet	Original paper [2012]	256x256
<input type="radio"/> GoogLeNet	Original paper [2014]	224x224

Nota: como se puede observar en la imagen, se muestra el formato de imagen aceptado por cada red, así como una referencia al Paper de la misma.

Para nuestro ejemplo, seleccionamos “AlexNet”.

Por último, indicamos el grupo y nombre de nuestro modelo:

Group Name ⓘ

Pruebas

Model Name ⓘ

Model_Pruebas

Create

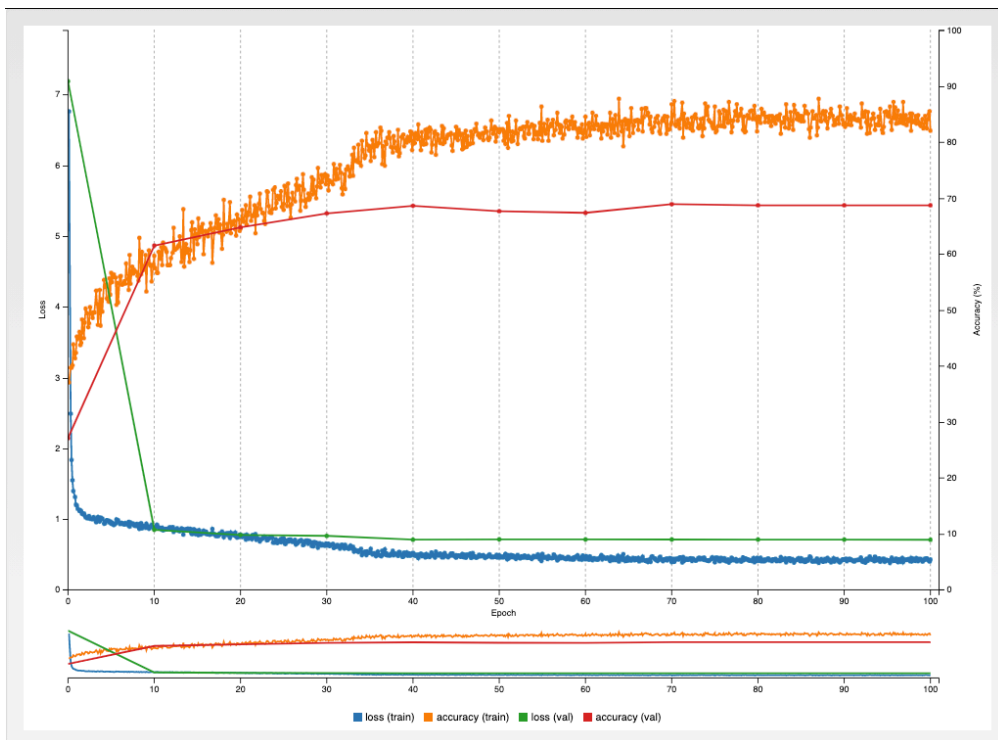
Y para finalizar, pulsamos en “Create”.

A partir de ahora, DIGITS pone a entrenar el modelo, proceso que puede llevar horas dependiendo del tamaño del dataset.

Notas Adicionales: se puede parar el entrenamiento del modelo pulsando en “Abort Job”. También se puede eliminar en “Delete Job” (automáticamente para el entrenamiento si este está en marcha) e incluso clonar con “Clone Job”.

Mientras se va entrenado, se puede ir viendo con todo detalle en la gráfica que va generando la precisión (accuracy) y las pérdidas (loss) tanto para entrenamiento como para validación.

Una vez entrenado, nos deberá salir una grafica similar a esta:



Podemos observar que el modelo de ejemplo tiene una precisión del 84,17 % para los datos de entrenamiento con unas pérdidas de 0.43 y una precisión del 68,75 % para los datos de validación con unas pérdidas de 0.71. Lo ideal es que la precisión sea la más alta posible y las pérdidas sean mínimas.

Para mejorar estos resultados, simplemente hay que jugar a modificar los parámetros al crear los modelos y personalizar las redes para adaptarlas al problema de clasificación que estamos intentando resolver.

Por último y no menos importante, podemos marcar el modelo como pre-entrenado pulsando sobre el botón "Make Pretrained Model". También podemos descargar el modelo pulsando sobre el botón "Download Model".

Probar Modelo

Para probar el modelo en DIGITS, una vez entrenado, tenemos una serie de opciones que se muestran a continuación:

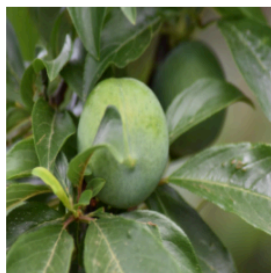
Tenemos dos opciones en líneas generales:

- **Test a single image:** probar una única imagen.
- **Test a list of images:** probar una lista de imágenes

Probar una única imagen

Para probar el modelo con una sola imagen, pulsamos en “Browse...” para seleccionar una imagen de nuestro ordenador, marcamos la opción “Show visualizations and statistics” y pulsamos en “Classify One”. Una vez clasificado, debe aparecer una pantalla similar a la siguiente dónde nos indica la predicción que ha hecho nuestro modelo:

Model_Pruebas_TF Image Classification Model



Description

Predictions

BlackDiamond 2019	88.77%
Angeleno Verde 2019	9.89%
RedBeaut Completo 2019	1.34%

Statistics

Debajo de esto, se puede observar de forma detallada la estructura de la red neuronal.

[Probar un conjunto de imágenes](#)

Para probar un conjunto de imágenes, tenemos las siguientes opciones:

Test a list of images

Upload Image List

Accepts a list of filenames or urls (you can use your val.txt file)

Image folder (optional)

Relative paths in the text file will be prepended with this value before reading

Number of images use from the file

Leave blank to use all

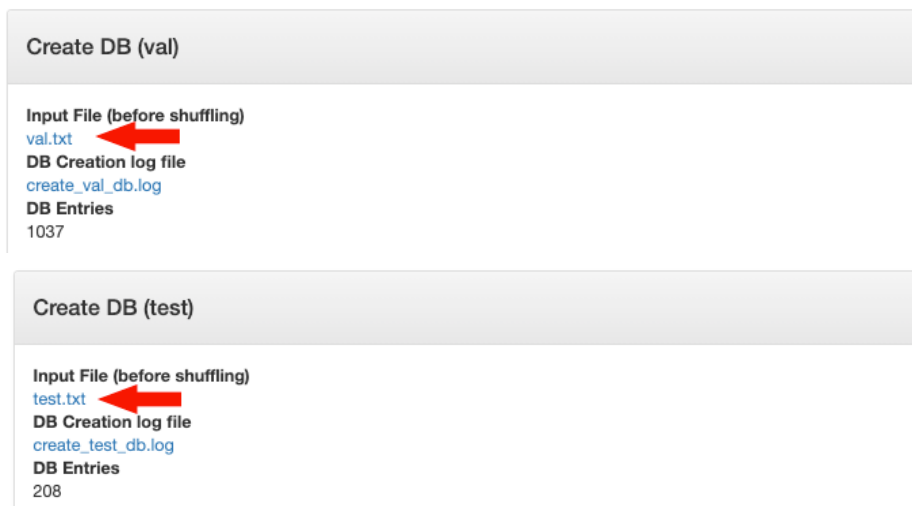
?

Number of images to show per category

?

- **Upload Image List:** deberemos subir un fichero de texto que contenga el listado de rutas a las imágenes a testear.

Recomendamos usar los ficheros de los conjuntos de validación y test que permite descargar DIGITS. Para ello, en una pestaña nueva del nuestro navegador web, abrimos la página de inicio de DIGITS (dirección de la máquina DIGITS), pulsamos sobre nuestro dataset dentro de la pestaña "Dataset". Una vez dentro del dataset, podemos descargar el fichero con el listado de rutas a las imágenes tanto para el conjunto de validación "Create DB (val)" como para el conjunto de datos de pruebas "Create DB (test)". **Nota:** recomendamos hacer uso del conjunto de imágenes de pruebas (test).



Nota Importante: para descargar el fichero, deberemos pulsar con el botón derecho del ratón en “Guardar enlace como...” y lo guardamos en nuestro ordenador.

- **Number of images use from the file:** indicar el número de imágenes a utilizar del fichero. Por defecto All (todas)

Para iniciar la prueba del modelo, marcamos la opción “Show visualizations and statistics” y pulsamos en “Classify many”. **Nota:** este proceso puede demorar bastante tiempo en completarse dependiendo del número de imágenes a clasificar.

Una vez clasificadas, debe aparecer una pantalla similar a la siguiente:

Model_Pruebas_TF Image Classification Model

Summary

Top-1 accuracy
69.71%

Top-5 accuracy
100.0%

Confusion matrix

	Angeleno Verde 2019	BlackDiamond 2019	RedBeaut Completo 2019	Per-class accuracy
Angeleno Verde 2019	33	8	15	58.93%
BlackDiamond 2019	15	45	13	61.64%
RedBeaut Completo 2019	6	6	67	84.81%

Job Status Done

- Initialized at 12:35:17 PM (1 second)
- Running at 12:35:18 PM (34 seconds)
- Done at 12:35:52 PM
(Total - 35 seconds)

Infer Model Done

Notes

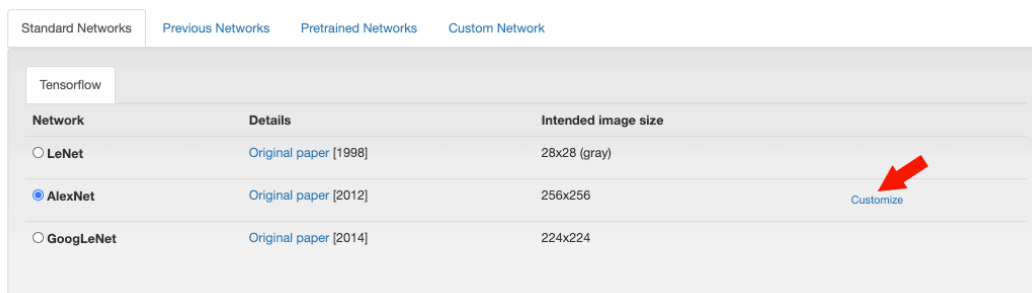
None

Como podemos observar en la imagen, nos muestra el resumen y la matriz de confusión. También nos muestra el listado completo con las estadísticas de clasificación de cada imagen.

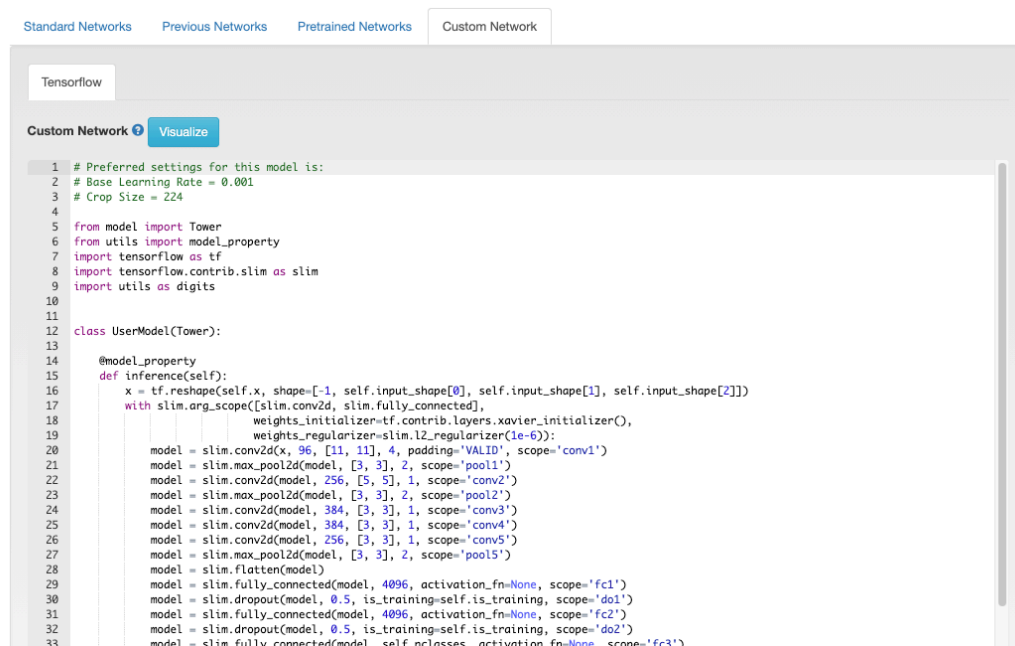
Personalizar Red Neuronal en DIGITS

Al crear un nuevo modelo, DIGITS permite la posibilidad de personalizar las redes neuronales.

Como se explicó en el apartado [Creación del Modelo de Clasificación](#), al seleccionar la red, aparece la opción “Customize” a la derecha tal como se muestra en la siguiente captura:



Si pulsamos, se nos abrirá el siguiente editor:



Como podemos observar, se muestra el código (en este caso TensorFlow) de la red y las diferentes capas que la componen. Para personalizar la red, simplemente hay que modificar el código en este editor.

En el ejemplo de la imagen (AlexNet) se puede observar que la red tiene 5 capas convolucionales (más dos capas “Max Pooling”) y 3 capas “Fully Connected”:

- **Capa Convolutiva:** conv2d (input, num_outputs, kernel_size, scope)
 - input: entradas (Tensor of rank N+2 of shape)
 - num_outputs: número de filtros (kernels)
 - stride: desplazamiento en n píxeles al aplicar los filtros
 - kernel_size: tamaño del filtro (kernel) en píxeles. Formato [n m]

- scope: nombre de la capa (opcional)

Más información:

<https://www.kite.com/python/docs/tensorflow.contrib.slim.conv2d>

- **Capa Max Pooling:** max_pool2d (input, kernel_size, stride, scope)
 - input: entradas (4-D tensor of shape)
 - stride: desplazamiento en n píxeles al aplicar los filtros
 - kernel_size: tamaño del filtro (kernel) en píxeles. Formato [n m]
 - scope: nombre de la capa (opcional)
- **Capa Fully Connected:** fully_connected (input, num_outputs, activation_fn, scope)
 - input: entradas
 - num_output: número de salidas (conexiones)
 - activation_fn: función de activación
 - scope: nombre de la capa (opcional)

Más información:

https://www.kite.com/python/docs/tensorflow.contrib.slim.fully_connected

Existen mas tipos de capas, métodos y operadores que se pueden consultar en <https://www.kite.com/python/docs/tensorflow.contrib.slim>.

Para más información: https://www.tensorflow.org/tutorials/customization/custom_layers.

Para visualizar la estructura de la red (únicamente DIGITS Caffe), pulse en “Visualize”.

Una vez que haya establecido todos los parámetros del modelo a entrenar, pulse en “Create” para iniciar el entrenamiento.

Solución de Problemas / Preguntas Frecuentes

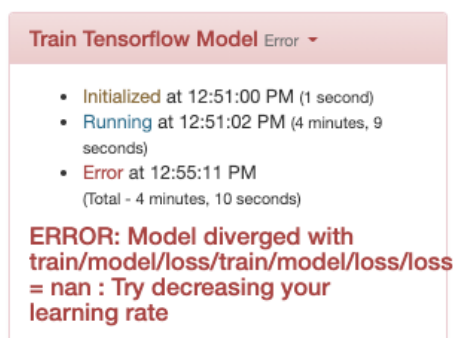
Filezilla no es capaz de conectarse con la máquina DIGITS

En primer lugar, compruebe que tiene una conexión a internet activa. En caso afirmativo, a continuación, verifique la configuración de la conexión a la máquina en Archivo -> Gestor de Sitios. En caso de que la configuración introducida sea la correcta y siga sin conectarse a la máquina, póngase en contacto con el responsable de la máquina.

No soy capaz de acceder a DIGITS desde el navegador

En primer lugar, verifique que ha introducido correctamente la dirección proporcionada en la barra de dirección de su navegador web. En caso de ser así, contacte con el responsable de la máquina.

Error “Model diverged with train/model/loss/train/model/loss/loss = nan : Try decreasing your learning rate” al entrenar el modelo con DIGITS TensorFlow



Para solucionar este error, debe pulsar en “Clone Job” clonar el modelo y establecer un Learning Rate (tasa de aprendizaje) menor. Para finalizar, elimine el modelo que ha dado errores con “Delete Job”.