

# Inference Test

```
In [1]: # all imports

import sys
import os
sys.path.insert(0, os.path.abspath('..'))
sys.path.insert(0, os.path.abspath('..\scripts'))

from fastai.vision.widgets import *
from fastai.interpret import ClassificationInterpretation
from fastbook import *
from helpers import *
import cv2

import torch
import torchvision
from torchvision.models import resnet50, ResNet50_Weights
```

```
In [2]: # get filename from models folder
learner_filename = input()
learner_filename = learner_filename.replace('.pth', '')
learner_filename
```

```
Out[2]: 'resnet_50_2023May08_0614PM'
```

```
In [3]: learner_path = os.path.join(get_models_dir(), learner_filename)
learner_path
```

```
Out[3]: 'D:\\ENGINEERING\\mezcal\\models\\resnet_50_2023May08_0614PM'
```

```
In [4]: # Load fastai Learner object
data_loader = make_data_loader(get_processed_data_dir(),
                               batch_size=64)
learn = resnet_learner(data_loader, 50)
learn.load(learner_path)
```

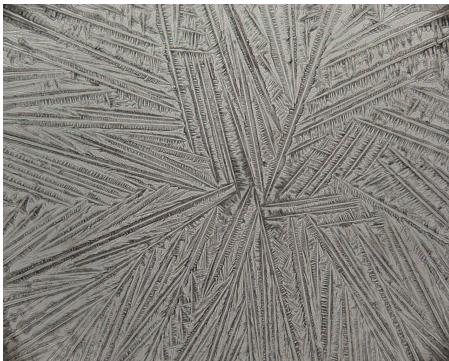
```
D:\\ENGINEERING\\mezcal\\venv\\lib\\site-packages\\torchvision\\models\\_utils.py:208: UserWarning: The parameter 'pretrained' is deprecated since 0.13 and will be removed in 0.15, please use 'weights' instead.
    warnings.warn(
D:\\ENGINEERING\\mezcal\\venv\\lib\\site-packages\\torchvision\\models\\_utils.py:223: UserWarning: Arguments other than a weight enum or `None` for 'weights' are deprecated since 0.13 and will be removed in 0.15. The current behavior is equivalent to passing `weights=ResNet50_Weights.IMAGENET1K_V1`. You can also use `weights=ResNet50_Weights.DEFAULT` to get the most up-to-date weights.
    warnings.warn(msg)
<fastai.learner.Learner at 0x21660d39fa0>
```

## fastai Inference Test

# Iterating through test folder

```
In [36]: from IPython.display import Image  
  
test_data_dir = get_test_data_dir()  
  
for fname in os.listdir(test_data_dir):  
    image_path = os.path.join(test_data_dir, fname)  
  
    image_object = cv2.imread(image_path)  
  
    prediction,_,probs = learn.predict(image_object)  
    print(f'{max(100*probs):.2f}% {prediction} - {image_path}')  
    image = Image(filename=image_path, width=224)  
    display(image)
```

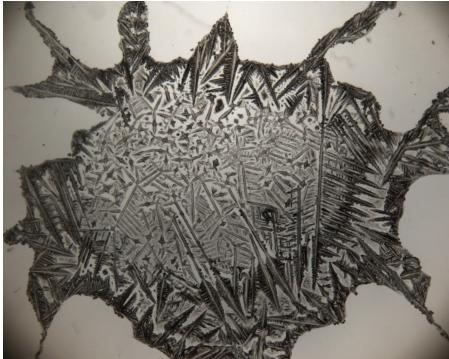
58.46% chino\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA A\_1.jpg



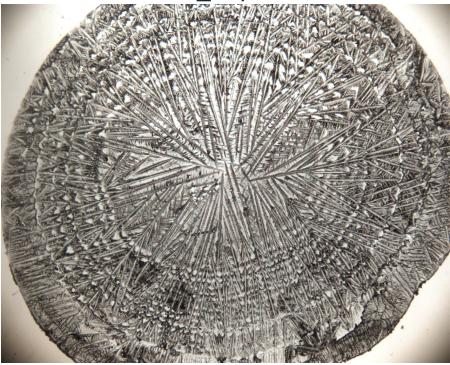
67.26% chino\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA A\_2.jpg



94.16% espadin\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA B.jpg



65.13% tobala\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA C.jpg



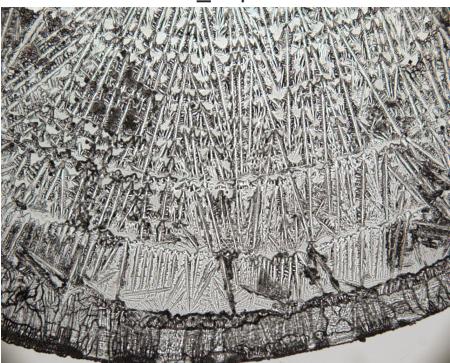
97.26% tobala\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA C\_2.jpg



100.00% tobala\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA C\_3.jpg



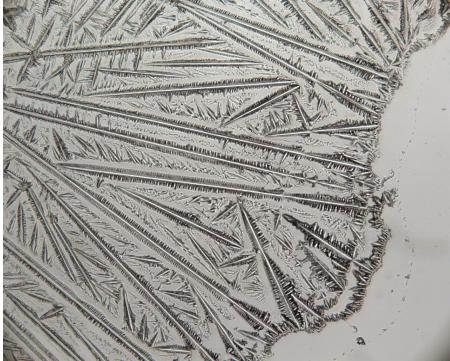
99.98% tobala\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA C\_4.jpg



99.75% espadin\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA D.jpg



97.10% espadin\_impure - D:\ENGINEERING\mezcal\data\test\MUESTRA D\_2.jpg



92.11% espadin\_impure - D:\ENGINEERING\mezcal\data\test\MUESTRA E.jpg



99.33% espadin\_impure - D:\ENGINEERING\mezcal\data\test\MUESTRA F.jpg



99.99% espadin\_impure - D:\ENGINEERING\mezcal\data\test\MUESTRA G.jpg



99.92% espadin\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRA H.jpg



44.29% chino\_impuro - D:\ENGINEERING\mezcal\data\test\MUESTRAA.jpg



```
In [6]: interp = ClassificationInterpretation.from_learner(learn)  
cm = interp.confusion_matrix()
```

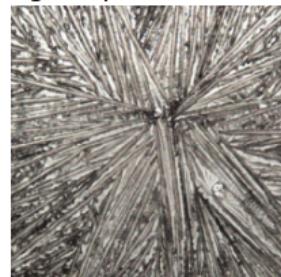
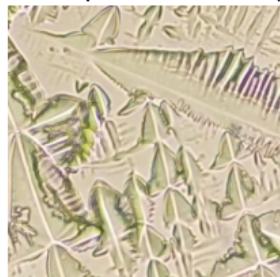
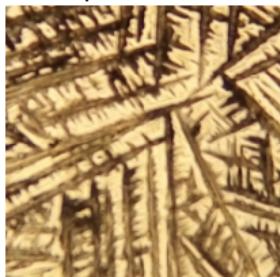
```
In [17]: interp.plot_confusion_matrix()
```

Confusion matrix									
Actual	background	chino_impuro	cuishe	espadin	espadin_impuro	pechuga	tepextate	tobala	tobala_impuro
background	4	0	0	0	0	0	0	0	0
chino_impuro	0	1	0	0	0	0	0	0	0
cuishe	0	0	13	0	0	1	0	0	0
espadin	0	0	0	14	0	0	1	0	0
espadin_impuro	0	0	0	0	12	0	0	0	0
pechuga	0	0	0	0	1	5	0	0	0
tepextate	0	0	0	0	0	0	5	0	0
tobala	0	0	0	3	0	0	0	17	0
tobala_impuro	0	0	0	0	1	0	0	0	7

```
In [7]: interp.plot_top_losses(9)
```

**Prediction/Actual/Loss/Probability**

cuishe/espadin / 6.09 / 0cuishe/espadin / 5.19 / 0.87  
pechuga/espadin / 2.73 / 0.40



espadin\_impuro/tobala\_impuro / 0.22 / 0.22 / 0.48  
tepextate / 1.45 / 0.148  
cuishe / 1.40 / 0.61



tobala/tepextate / 1.34 / 0.55  
tobala/pechuga / 1.27 / 0.66  
espadin/tobala / 1.27 / 0.43

