Introduction to Computer vision with Python

Pie and AI, Kochi

About ME

- Incoming Data Scientist-SE @ AOT Technologies
- Open source contributor - Keras, DVC
- Data science Enthusiast
- GCI Mentor(Tensorflow), 2019
- FOSSASIA Open Technights winner

Computer vision

Where does python come in picture?

Resources

Are there other languages?

- Yes / No question

Getting started in CV

Face Mask classifier

“ Talk is cheap. Show me the code. ”

Linus Torvalds

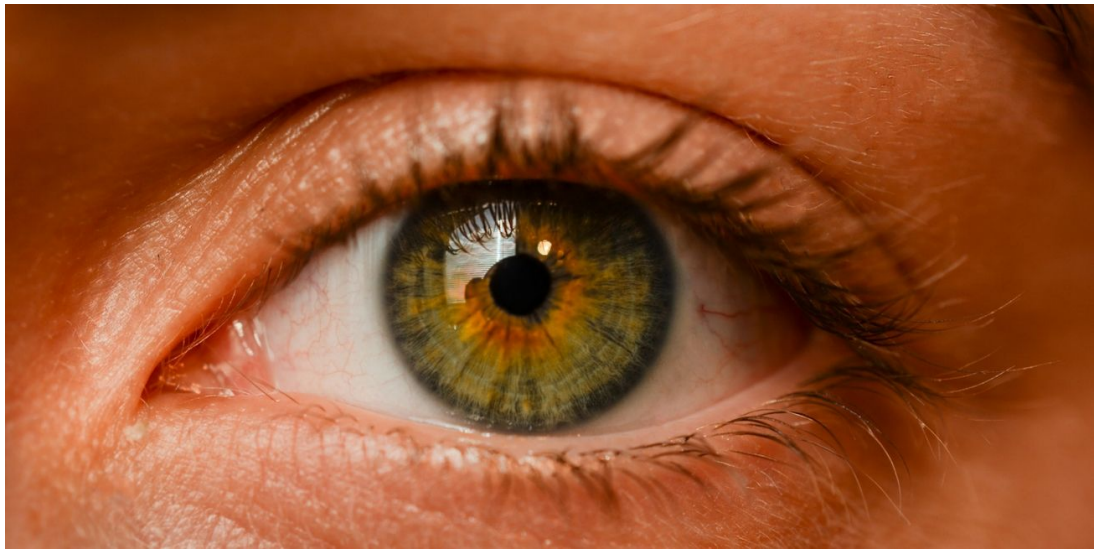
```
# Loading dataset
item_tfms = Resize(256)
batch_tfms = [RandomResizedCrop(224), *aug_transforms(mult=1.0,
do_flip=True, max_rotate=30.0, max_zoom=1.5,
                          max_lighting=.8, max_warp=0.3, p_lighting=.9)]
bs=128
data = ImageDataLoaders.from_folder(path, train='Train',
valid='Validation',batch_tfms=batch_tfms,
                                    item_tfms=item_tfms, bs=bs)


# Fastai cnn_Learner

learn = cnn_learner(data, resnet50, metrics=accuracy)
learn.fine_tune(3, cbs=[EarlyStoppingCallback(monitor='accuracy')])
```

Lets dive into code

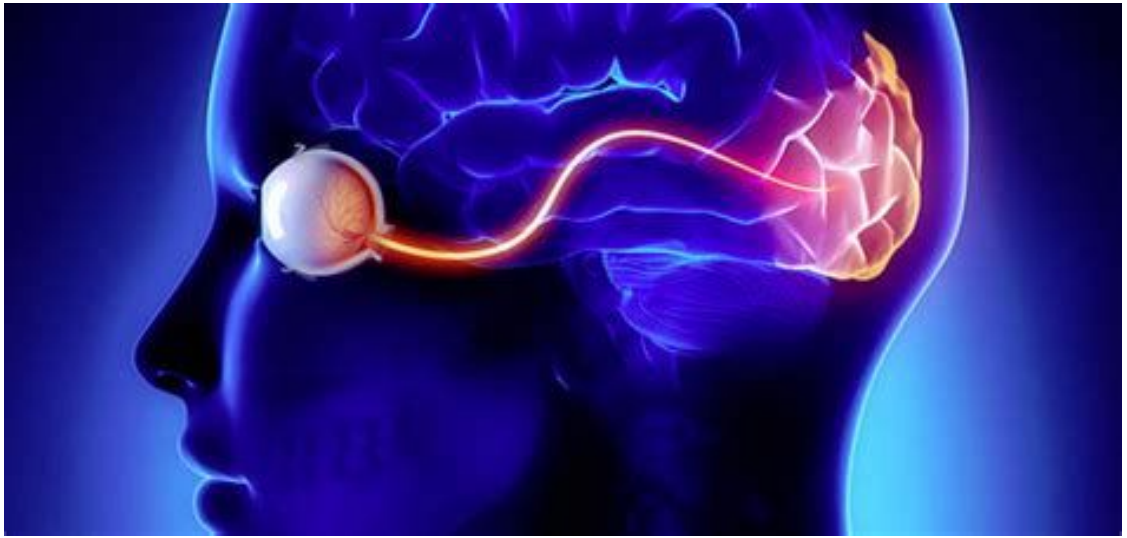
https://www.kaggle.com/kurianbenoy/with-or-without-facemast-fastai2/

Computer vision

- We humans are used for considering vision as like something by default.
- Lot of learning required and previous experiences shape our vision



Computer vision(CV)



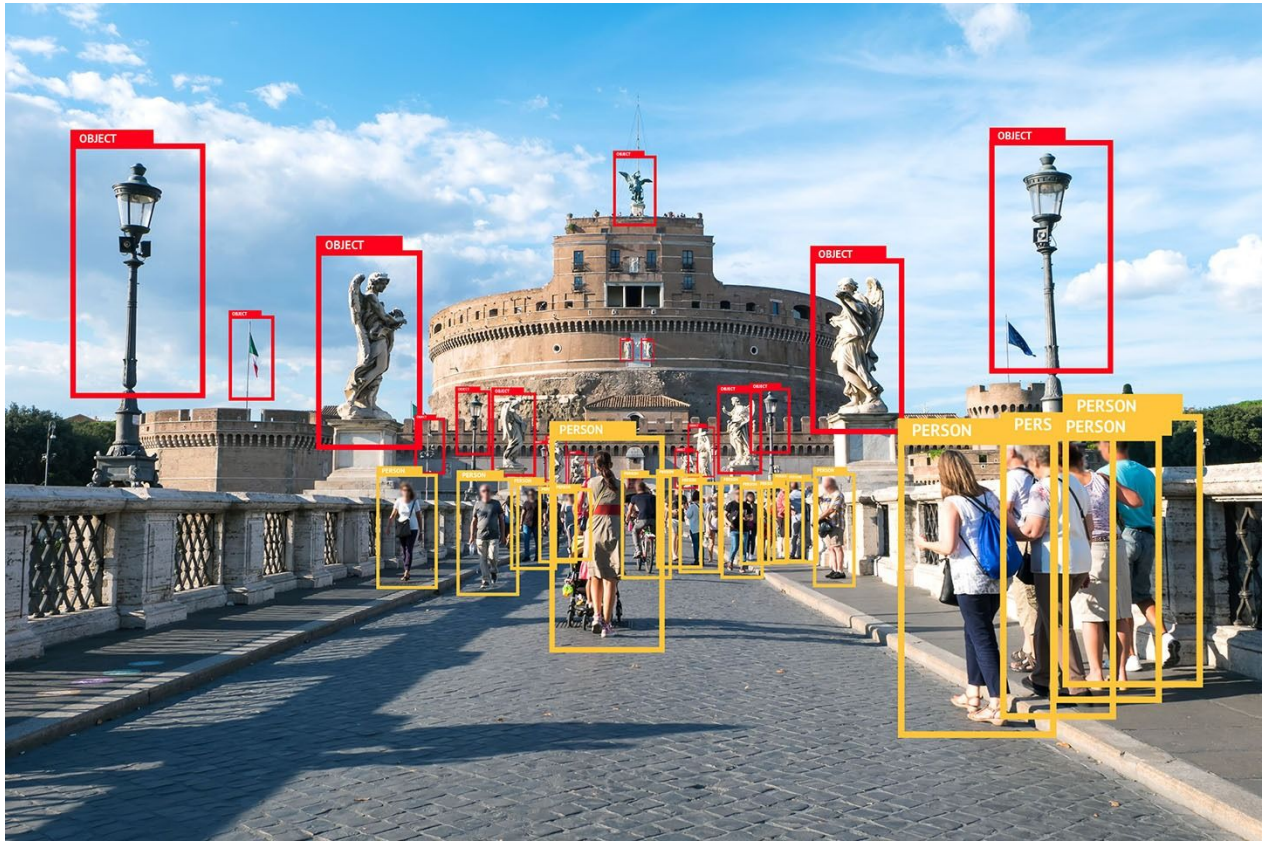
Visual begins with eyes but it truly takes place in the brain

- **Dr Fei Fei Li**

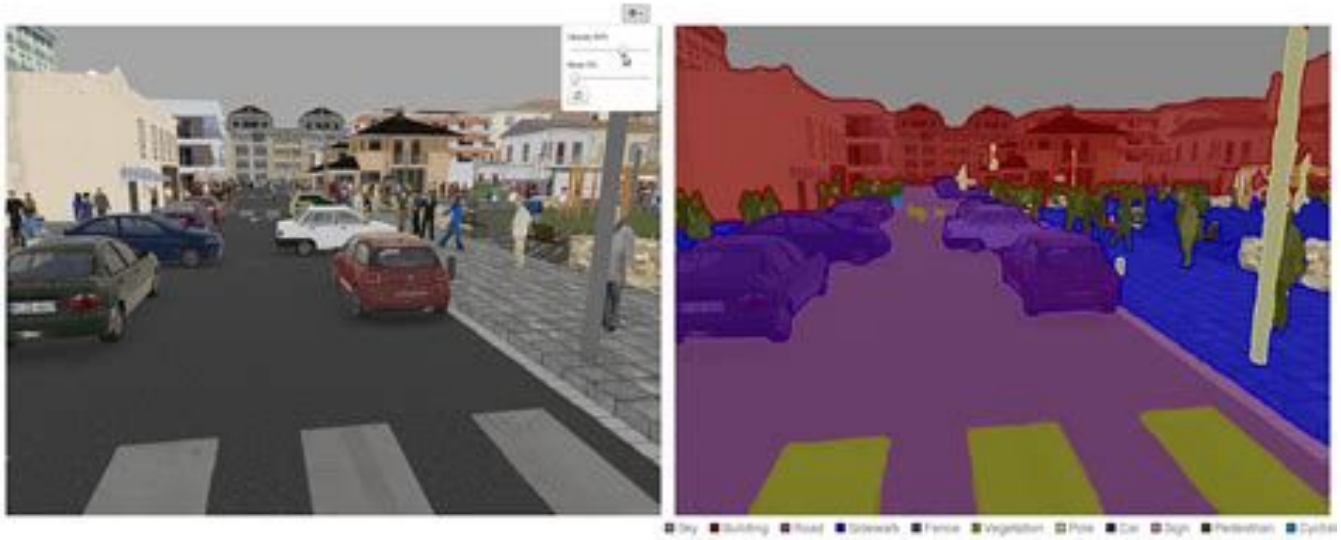
Computer vision tasks

Image Classification

Object Detection

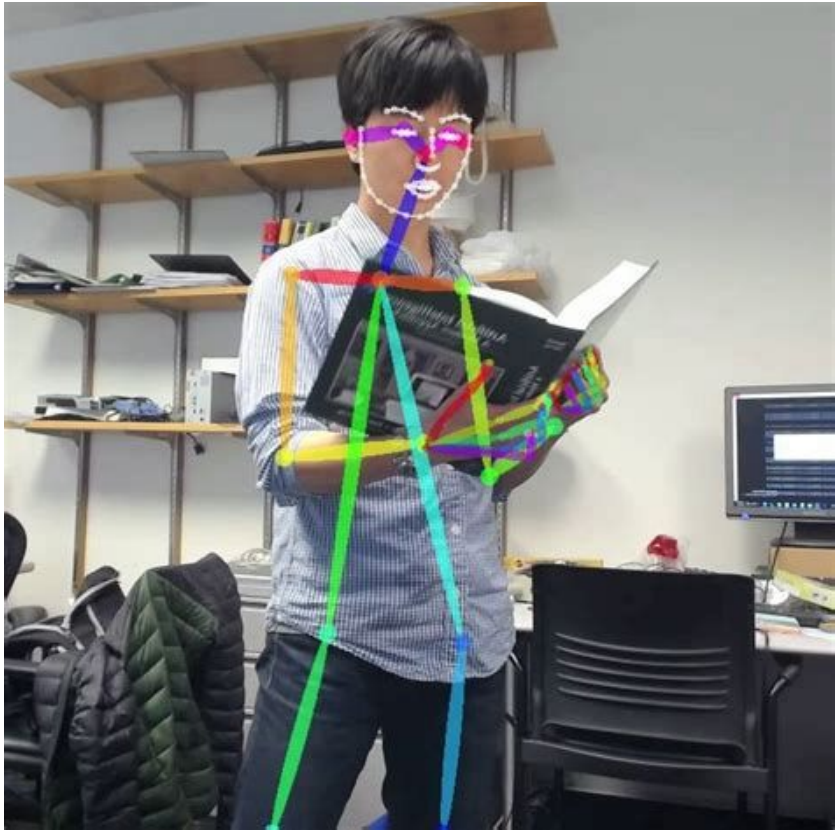


Segmentation

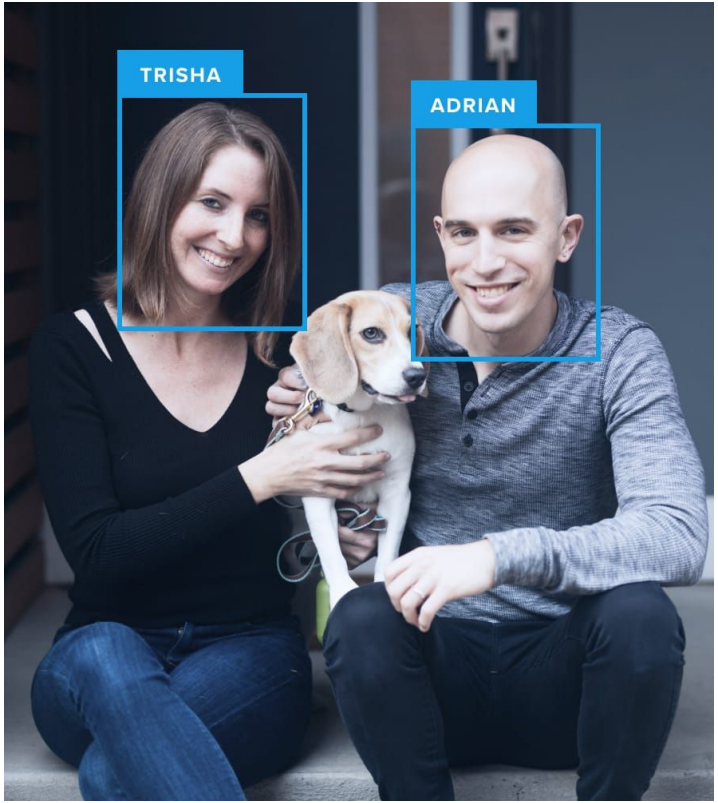


Activity Detection

Pose Detection



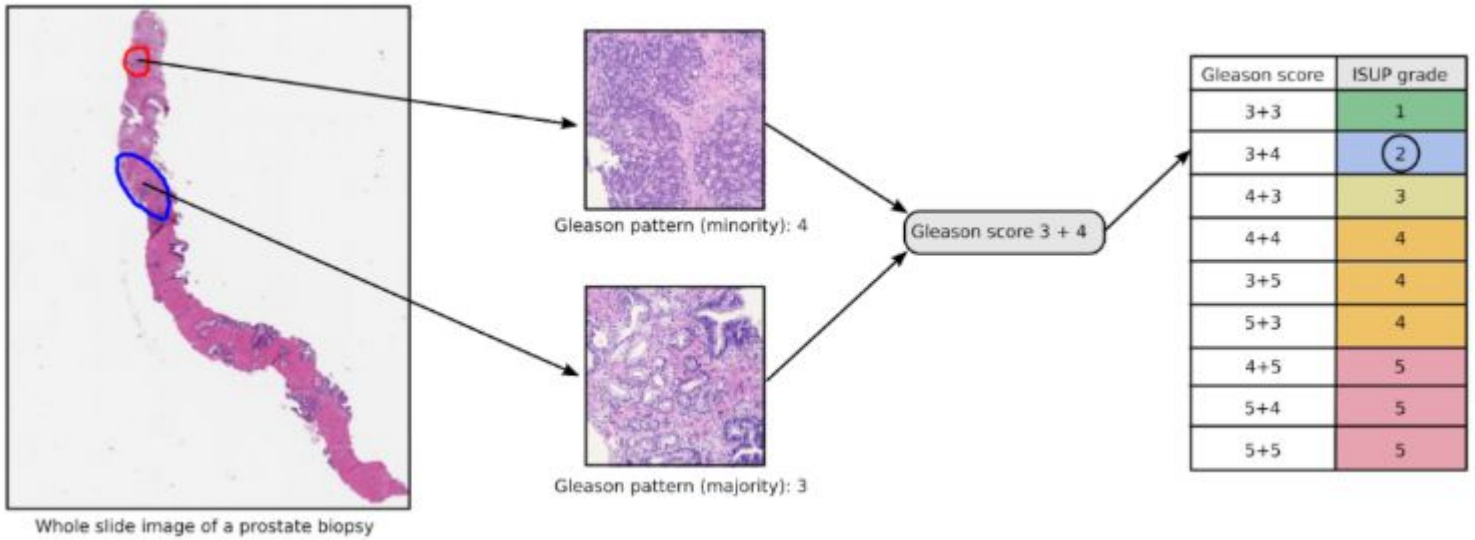
Person Recognition



Application

Estimating Air quality

Biotech AI


Whole slide image of a prostate biopsy
Gleason pattern (minority): 4
Gleason pattern (majority): 3
Gleason score 3 + 4
Gleason score
ISUP grade
3+3
1
3+4
2
4+3
3
4+4
4
3+5
4
5+3
4
4+5
5
5+4
5
5+5
5

Plant classifier

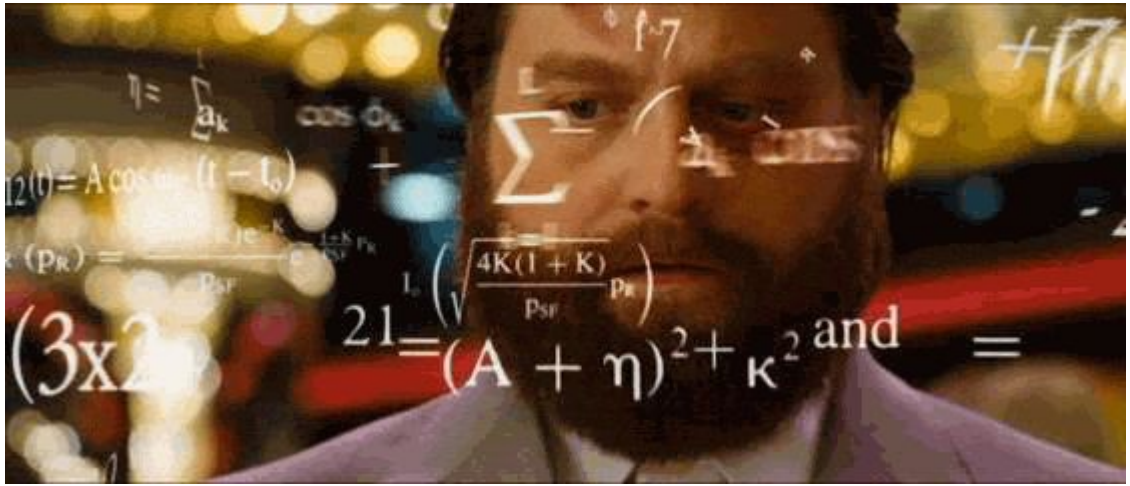
ART

Bus classifier

Image classification

Image classification

It’s the process of classifying a given image with a set of labels which is already being predefined.



With mask

Without mask

Steps

1. Taking a look at the data
2. Data preprocessing
3. Modelling our image classifier
4. Training our model
5. Evaluating our model

1. Taking a look at your data

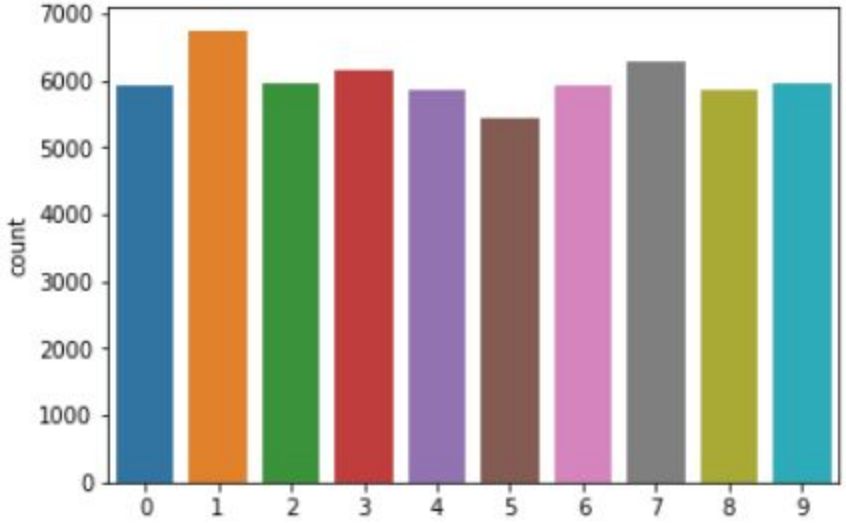
- Looking at images
- Understand the class balance
- Check for corrupted images, outliers in data

```
# Load the dataset using keras library

(x_train, y_train), (x_test, y_test) = keras.datasets.mnist.load_data()
```

```
# Load the dataset using keras library

(x_train, y_train), (x_test, y_test) = keras.datasets.mnist.load_data()


# Load train labels distribution

y = np.bincount(y_train)
y_count = np.nonzero(y)[0]
print(y_count)
print(y)
sns.countplot(y_train)
```

```
[0 1 2 3 4 5 6 7 8 9]
[5923 6742 5958 6131 5842 5421 5918 6265 5851 5949]
```

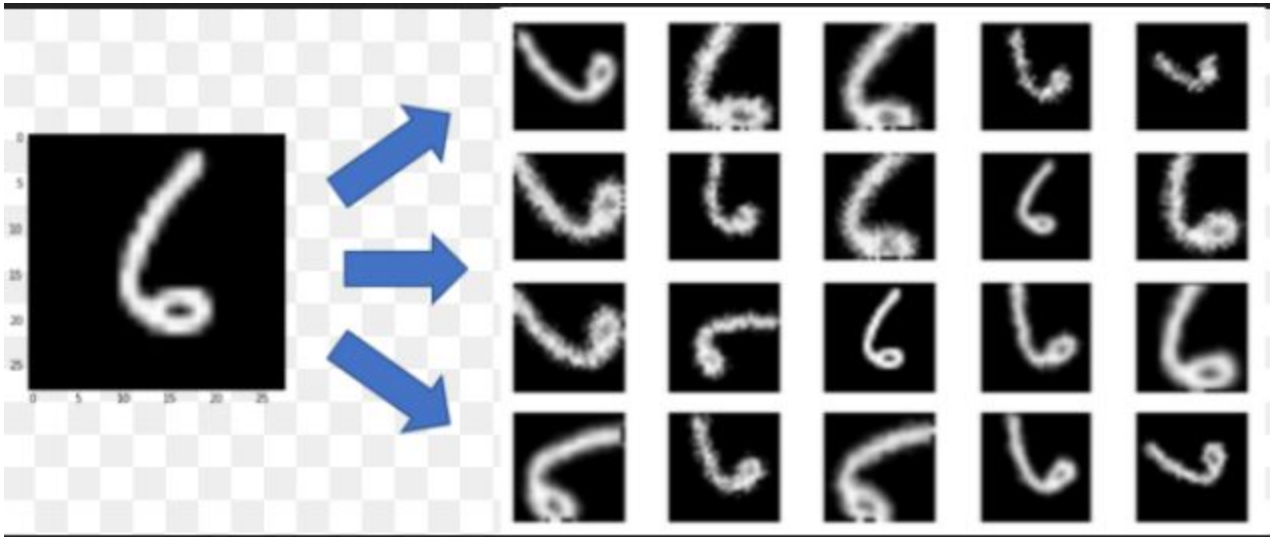
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f3bc29307b8>
```



2. Data Preparation

- Resizing images
- Preprocessing techniques like: Normalisation, reshape, label encoding
- Applying data augmentations

Data augmentation



3. Modelling our image classifier

Image classification - CNN

Not always

- KNN
- Image classification architectures

```
# Build a simple model
```

```
# Build a simple model
inputs = keras.Input(shape=(28, 28))


x = layers.Flatten()(inputs)
x = layers.Dense(128, activation="relu")(x)
x = layers.Dense(128, activation="relu")(x)
outputs = layers.Dense(10, activation="softmax")(x)
model = keras.Model(inputs, outputs)

model.summary()
```

Common image classification architectures

- VGG
- Resnet 18, 34, 50, 101
- MobileNet
- DenseNet
- EfficientNet (B0-B8)
- Big Transfer

Transfer learning

```
# Build a simple model
from tensorflow.keras.applications import Xception

## Load a convolutional base with pre-trained weights
base_model =Xception(
    weights='imagenet',
    include_top=False,
    pooling='avg')

# Freeze the base model
base_model.trainable = False
```

```
# Build a simple model
inputs = keras.Input(shape=(150, 150, 3))

# We make sure that the base_model is running in inference mode here,
# by passing `training=False`.

x = base_model(inputs, training=False)
x = keras.layers.GlobalAveragePooling2D()(x)
outputs = keras.layers.Dense(1)(x)

model = keras.Model(inputs, outputs)
```

4. Training our model

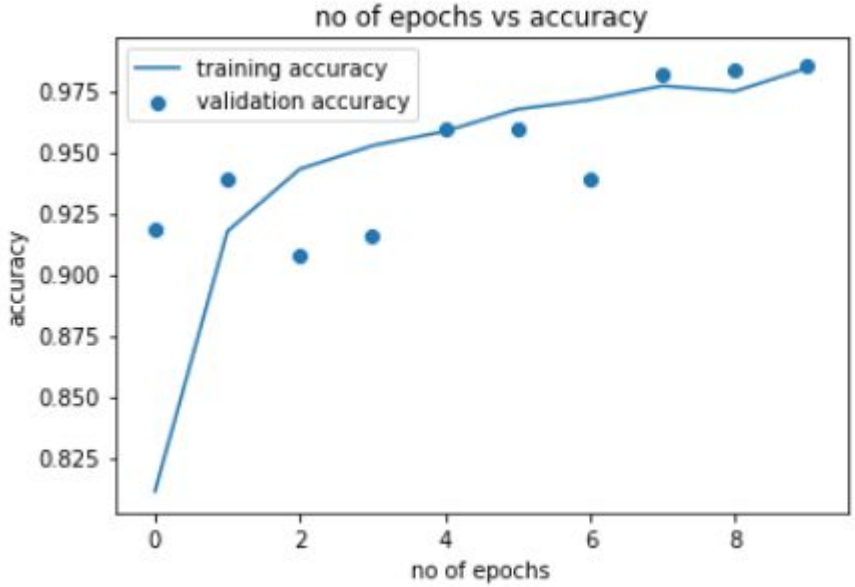
It’s usually calling:

model.compile(..)

model.fit(..)

- loss: we will set our loss as binary_crossentropy since we are attacking a binary classification problem
- optimizer : optimizers shape and mold your model into its most accurate possible form by futzing with the weights.
- metrics : This is the evaluation criteria that we choose to evaluate our model

```
# Use a Sequential model to add a trainable classifier on top
model.compile(optimizer='adam', loss='crossentropy', metrics=['accuracy'])

model.fit(dataset,steps_per_epoch=100,
    epochs=10,validation_data=validation_data,
    validation_steps=50)
```

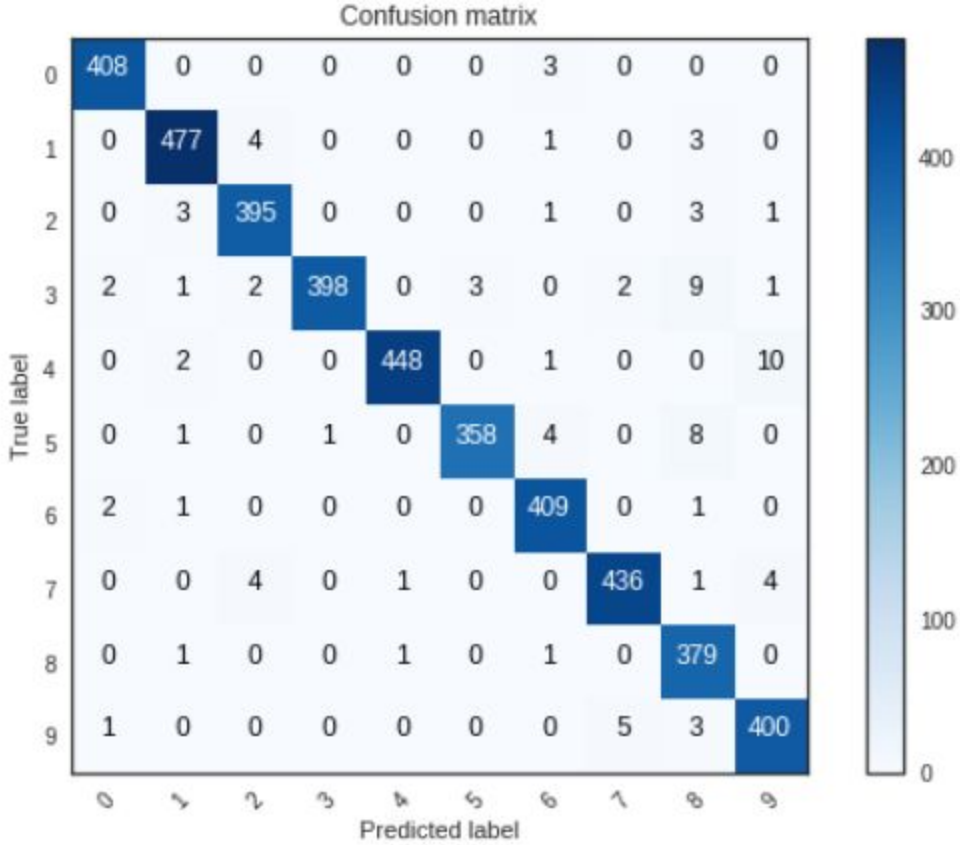

no of epochs vs accuracy
training accuracy
validation accuracy
accuracy
0.975
0.950
0.925
0.900
0.875
0.850
0.825
0
2
4
6
8
no of epochs

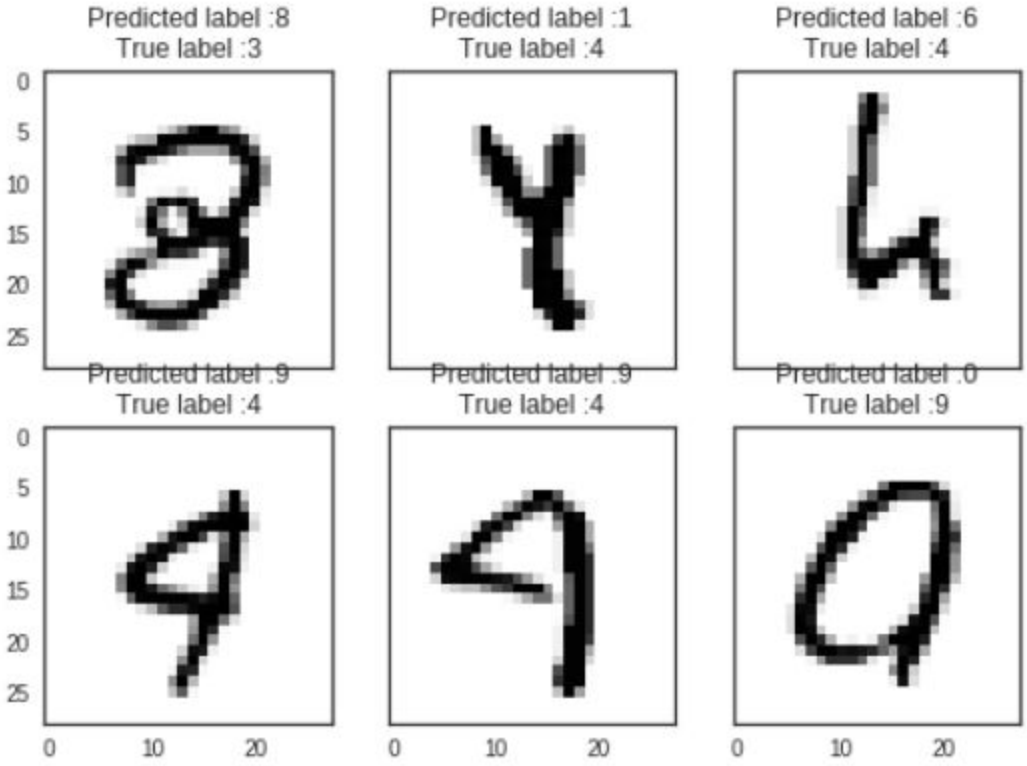
Tips for modelling

1. Start with small models(maybe stack a layers of convnets yourself)
2. Don’t use data preprocessing(augmentations initially)
3. Fix random seed
4. Get a dump baseline first
5. Then overfit

5. Evaluate results

- Set baselines
- Monitor your experiments
- Human evaluation


Confusion matrix
True label
Predicted label
0 1 2 3 4 5 6 7 8 9
408 0 0 0 0 0 3 0 0 0
0 477 4 0 0 0 1 0 3 0
0 3 395 0 0 0 1 0 3 1
2 1 2 398 0 3 0 2 9 1
0 2 0 0 448 0 1 0 0 10
0 1 0 1 0 358 4 0 8 0
2 1 0 0 0 0 409 0 1 0
0 0 4 0 1 0 0 436 1 4
0 1 0 0 1 0 1 0 379 0
1 0 0 0 0 0 0 5 3 400
400
300
200
100
0


Predicted label :8
True label :3
Predicted label :1
True label :4
Predicted label :6
True label :4
Predicted label :9
True label :4
Predicted label :9
True label :4
Predicted label :0
True label :9
0
5
10
15
20
25
0
10
20

Convolutional neural networks

Acknowledgment

Indian way to learn CNN - Shahul E.S

Keras getting started - fchollet

Let’s get connected on Twitter! I am [@kurianbenoy2](https://twitter.com/kurianbenoy2).