Deep Learning School - Galaxy Classification Challenge

Group 6: Oleksandr Koshchii, Alexander Pesch Berrocal, Peter-Bernd Otte, Sara Aumiller, Johanna Kraus

March 2, 2023

ERUM DATA HUB

Dataset

- The Dataset: Galaxy10
- only grayscale
- cropped to 64x64 pixels
- input images: 21 785

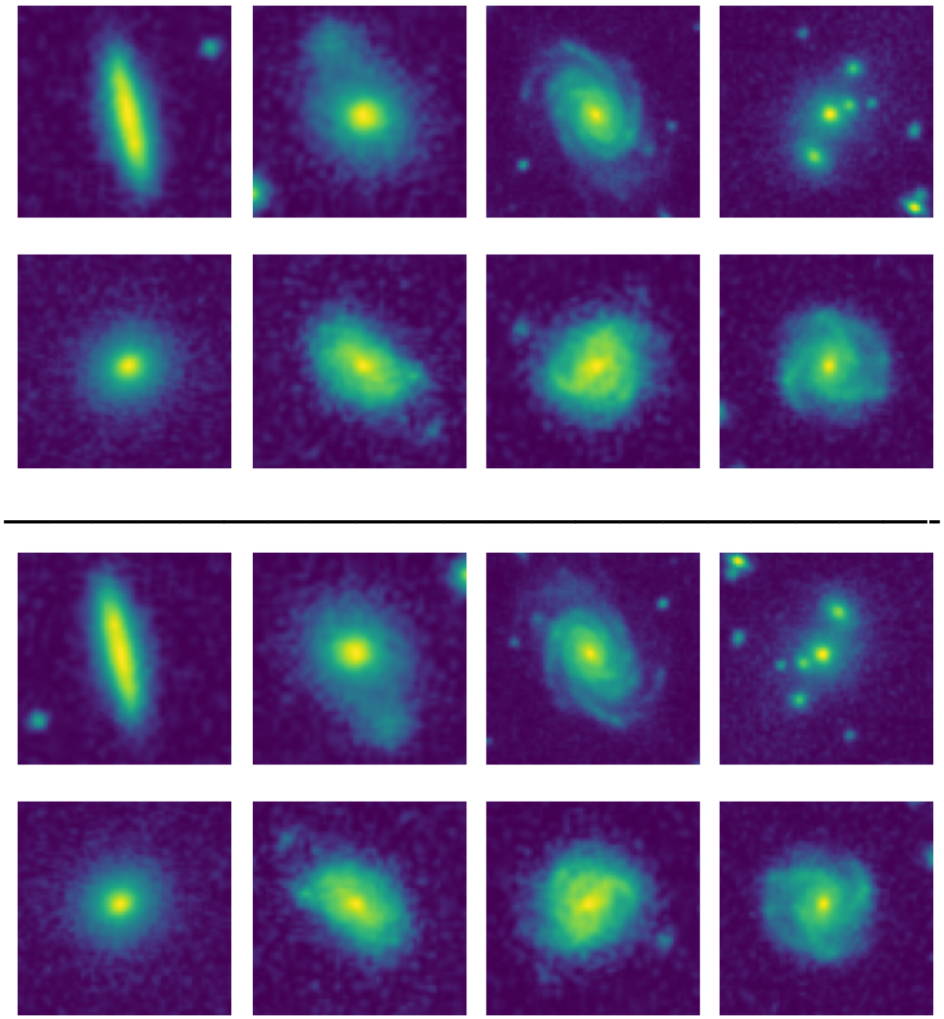
We have 10 classes:

0 'Disk, Face-on, No Spiral': 3461
1 'Smooth, Completely round': 6997
2 'Smooth, in-between round': 6292
3 'Smooth, Cigar shaped': 349
4 'Disk, Edge-on, Rounded Bulge': 1534
5 'Disk, Edge-on, Boxy Bulge': 17
6 'Disk, Edge-on, No Bulge': 589
7 'Disk, Face-on, Tight Spiral': 1121
8 'Disk, Face-on, Medium Spiral': 906
9 'Disk, Face-on, Loose Spiral': 519

- Data augmentation:
  - Increase the statistics: flip and rotate the dataset
    $\rightarrow$ 5 x more data (108 925 pictures)
  - Deal with imbalance of classes: Apply class weights: $w = N_{all}/N_{galaxy}$

Augmentation of the Dataset

- Performed up-down flip
- Performed left-right flip
- **Performed 180 degrees rotation**
- Performed 270 degrees rotation
- Combined dataset: 108925



Our Network

We are using **Tensorflow/Keras**

```
# training hyperparameters
learning_rate=0.0003
batch_size=50
epochs=20
layers = tf.keras.layers

# model definition
model = tf.keras.models.Sequential()

model.add(layers.Conv2D(filters=10, kernel_size=(8, 8), activation='ReLU', input_shape=(64,64,1), padding="valid", strides=2))
model.add(layers.MaxPool2D(pool_size=(3,3), strides=2))
model.add(layers.Conv2D(filters=64, kernel_size=(5,5), activation='ReLU', input_shape=(14, 14, 10), padding="same"))
model.add(layers.MaxPool2D(pool_size=(2,2), strides=1))
model.add(layers.Conv2D(filters=16, kernel_size=(3,3), activation='ReLU', input_shape=(13, 13, 64), padding="same"))
model.add(layers.Conv2D(filters=16, kernel_size=(3,3), activation='ReLU', input_shape=(13, 13, 16), padding="same"))
model.add(layers.MaxPool2D(pool_size=(2,2), strides=2))
model.add(layers.Flatten())

model.add(layers.Dense(units=64, activation = "relu"))
model.add(layers.Dropout(0.2))
model.add(layers.Dense(units=32, activation = "relu"))
model.add(layers.Dense(units=10, activation='softmax'))
model.summary()
```

- Sequential model with 67604 parameters
- Optimizer: Adam ($\alpha = 0.0003$)
- 20 epochs with categorical crossentropy loss
- validation split of 80:20 (87140, 21785)

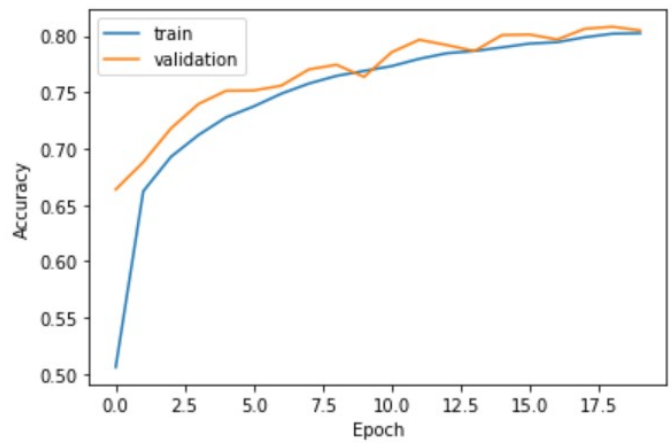
Performance

- Accuracy as most important measure

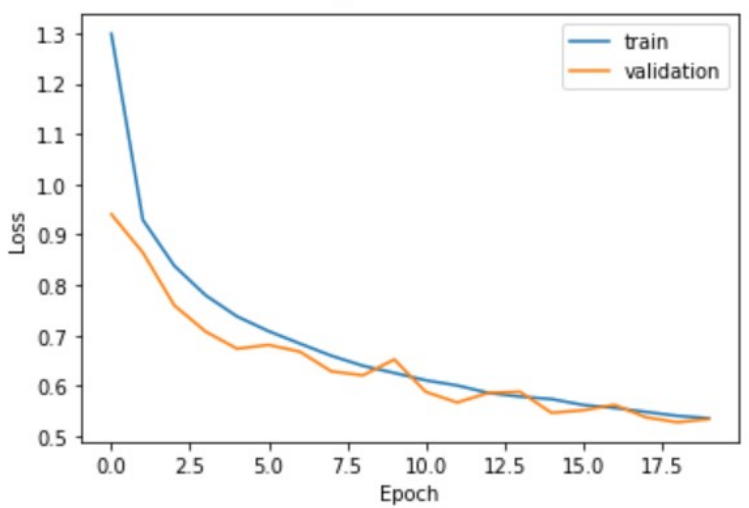
→ High accuracy: want to have the most galaxies right

- smooth learning rate with trainings loss ≈ validation loss

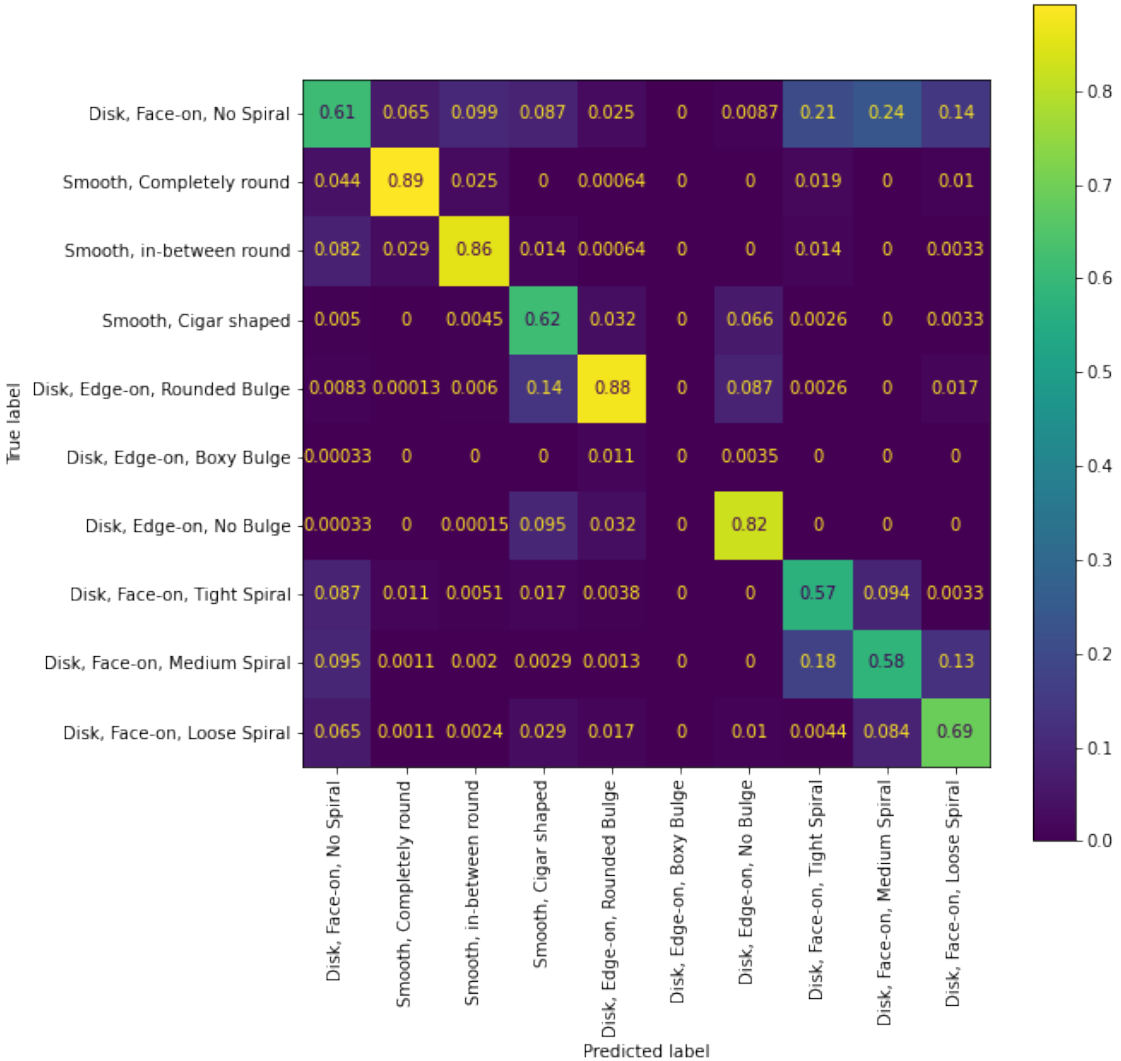
- No overtraining visible here




```
Best validation accuracy: 0.8082396388053894
```



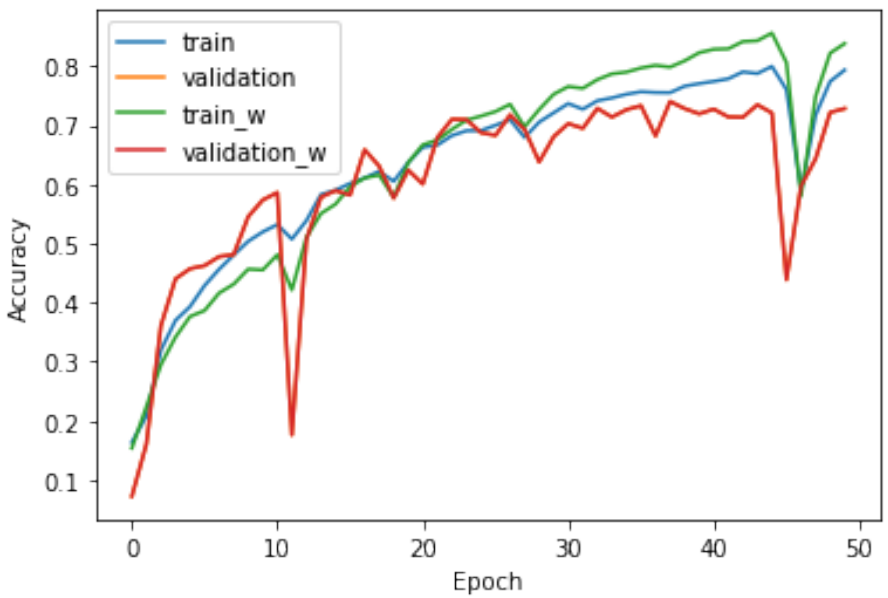
Confusion matrix (no class weights used yet)

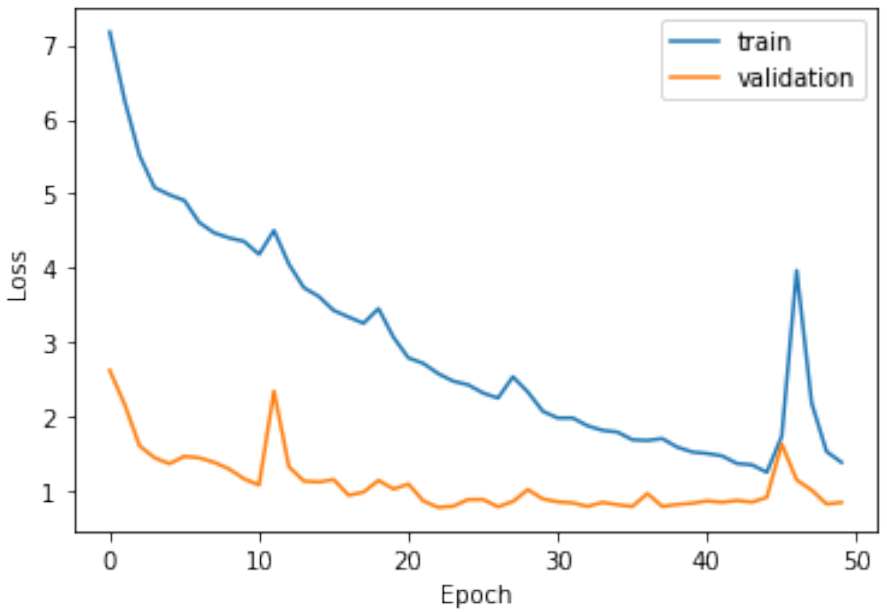


- Worst performance: galaxy 5: disk egde-on boxy bulge ← low statistics
- Best performance: galaxy 1, 2, 4 ← high statistics
- Hard to learn: different spirals of disk face-on galaxies (0, 7, 8, 9)

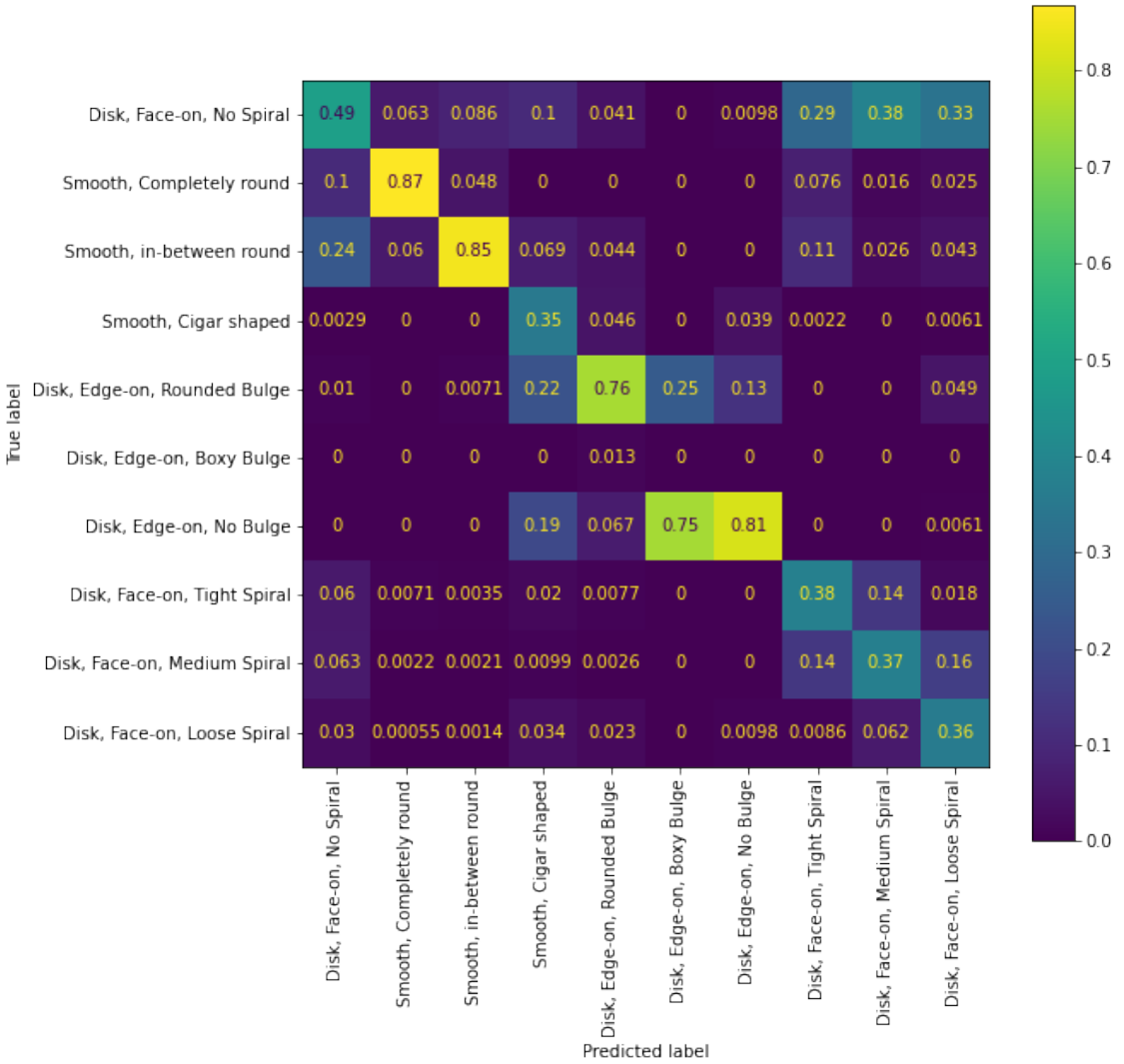
Performance with weights

- Idea: Compromise with lower accuracy but better classification of class 5
- smooth learning rate with trainings loss $\approx$ validation loss
- no dropout layer currently
- Best validation accuracy 84%





Performance with weights



- Does affect all classes!

Conclusions

- We successfully trained a network to identify galaxies
  $\rightarrow$ best validation accuracy: 80%
- Very good identification of smooth galaxies

- Room for improvement for class 5: low statistics