Face Mask Detection

Group 12

Andrei Cozma, Sabrullah Deniz, Hunter Price,
Isaac Sikkema, Nan Tang

Problem Motivation

- **Face-Mask Recognition**
  - Health-related concerns benefits
  - Commercial uses

Current State of the Art

- Computer Vision and Pattern Recognition
- Architectures & Pre-Trained Models
  - VGG19 Model, Xception Model, ConvNet, InceptionV3, MobileNet, MobileNetV2, DenseNet20

Challenge & Dataset





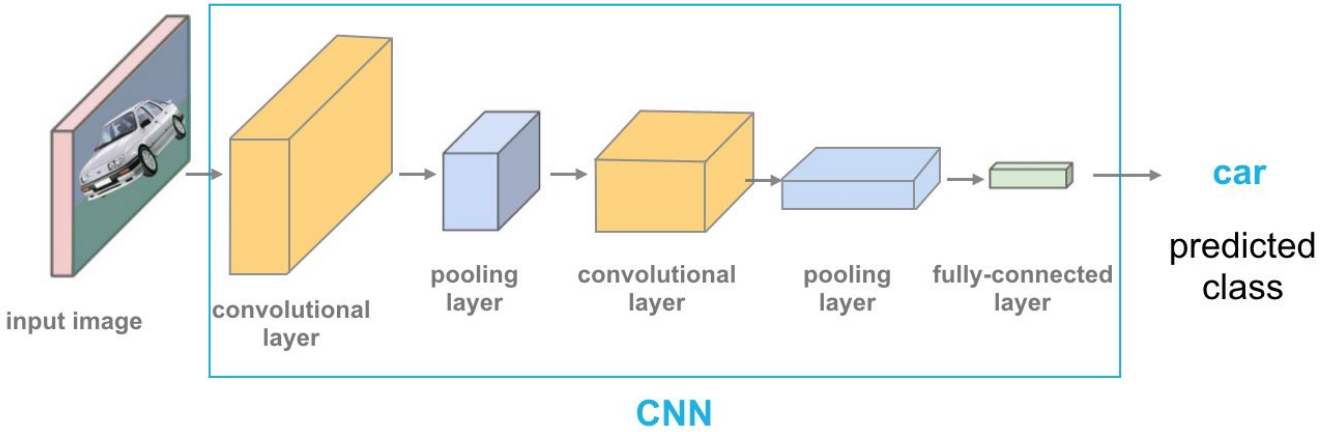



Pre-Processing

- Techniques employed
  - Resizing (128x128)
  - Grayscaling (single channel)
  - Min-Max Scaling (values between 0 and 1)
- Convolutional Network
  - Optimizing architecture
  - Enhanced results (+10% acc.)

Feature Extraction / Selection

- Convolutional Neural Network
  - Convolution Layers
  - Pooling Layers
  - Activation Layers



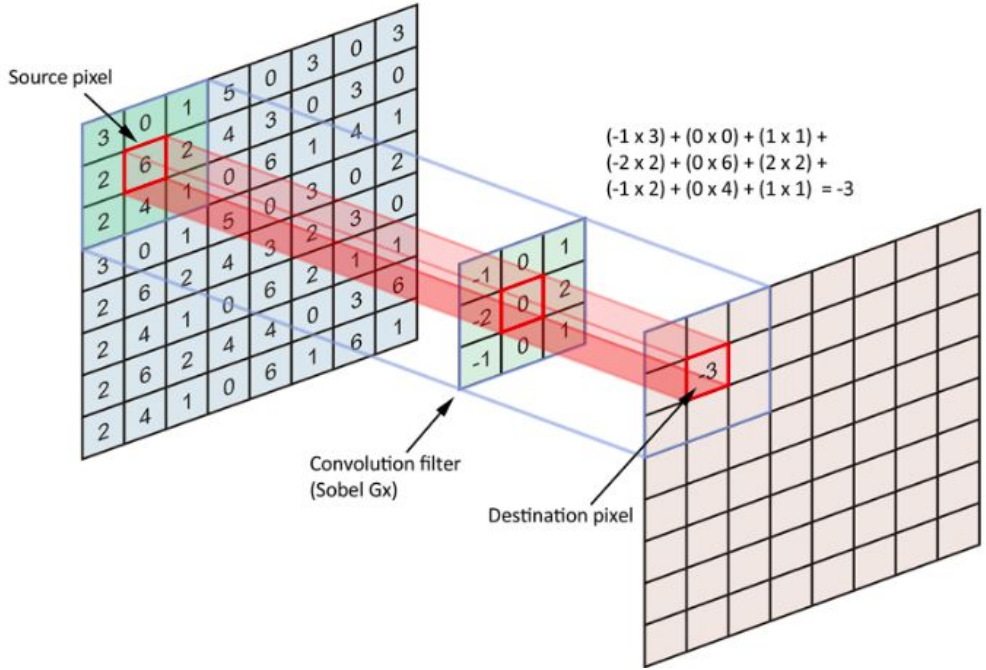
Convolution

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

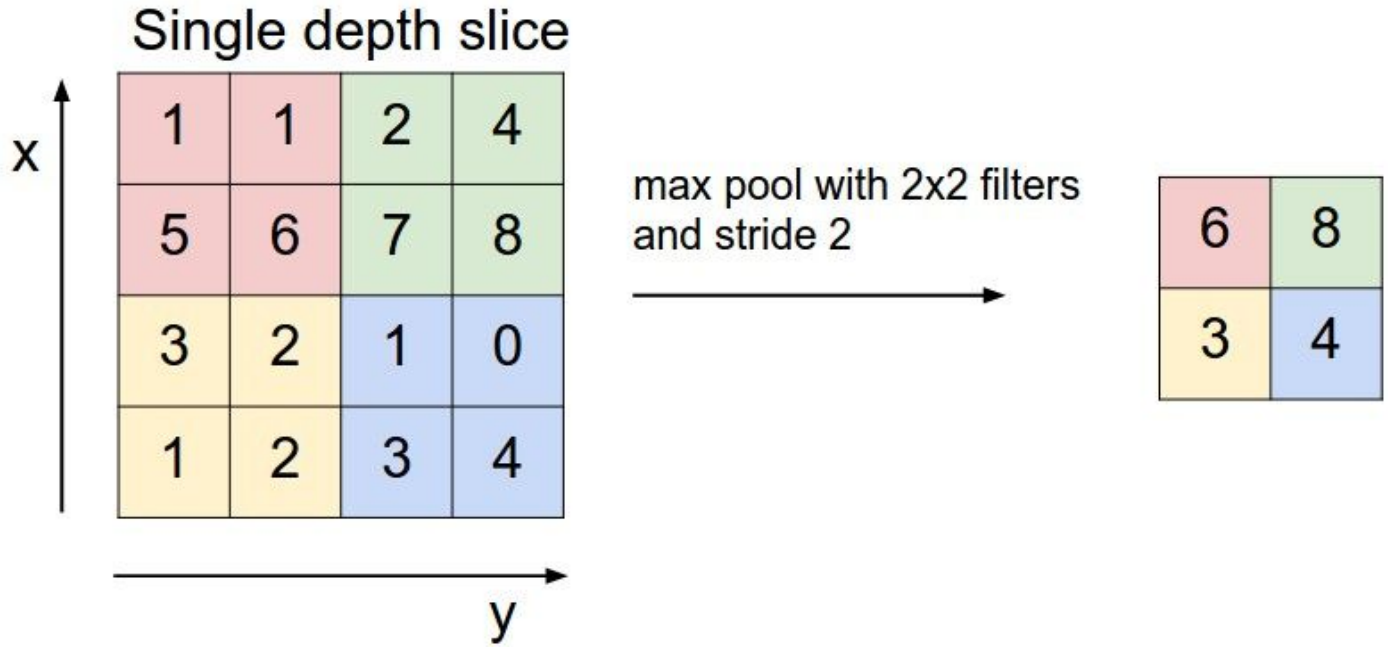
Outline Filter

$$\begin{bmatrix} -1 & -1 & -1 \\ 1.15 & 1.15 & 1.15 \\ -1 & -1 & -1 \end{bmatrix}$$

Horizontal Line Filter

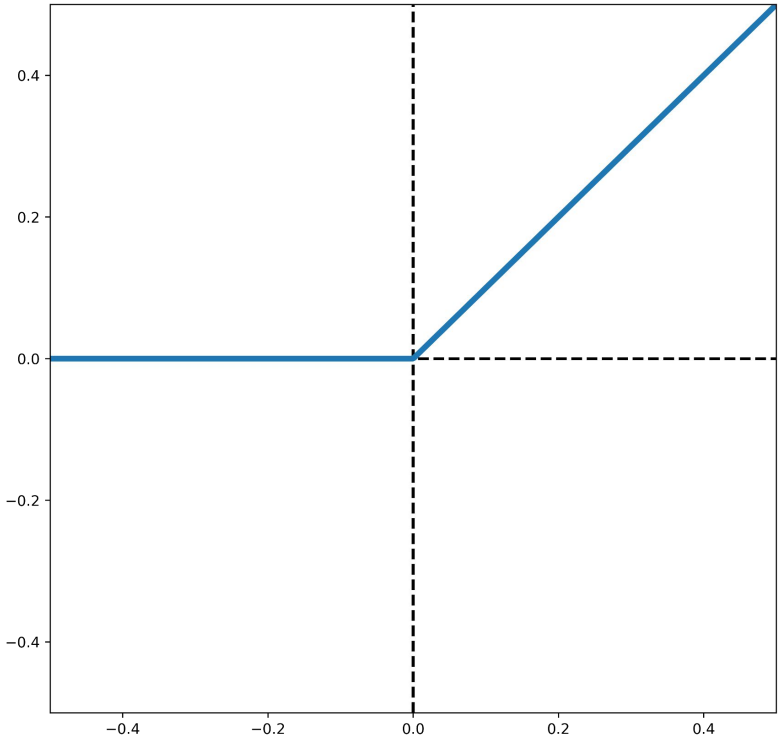


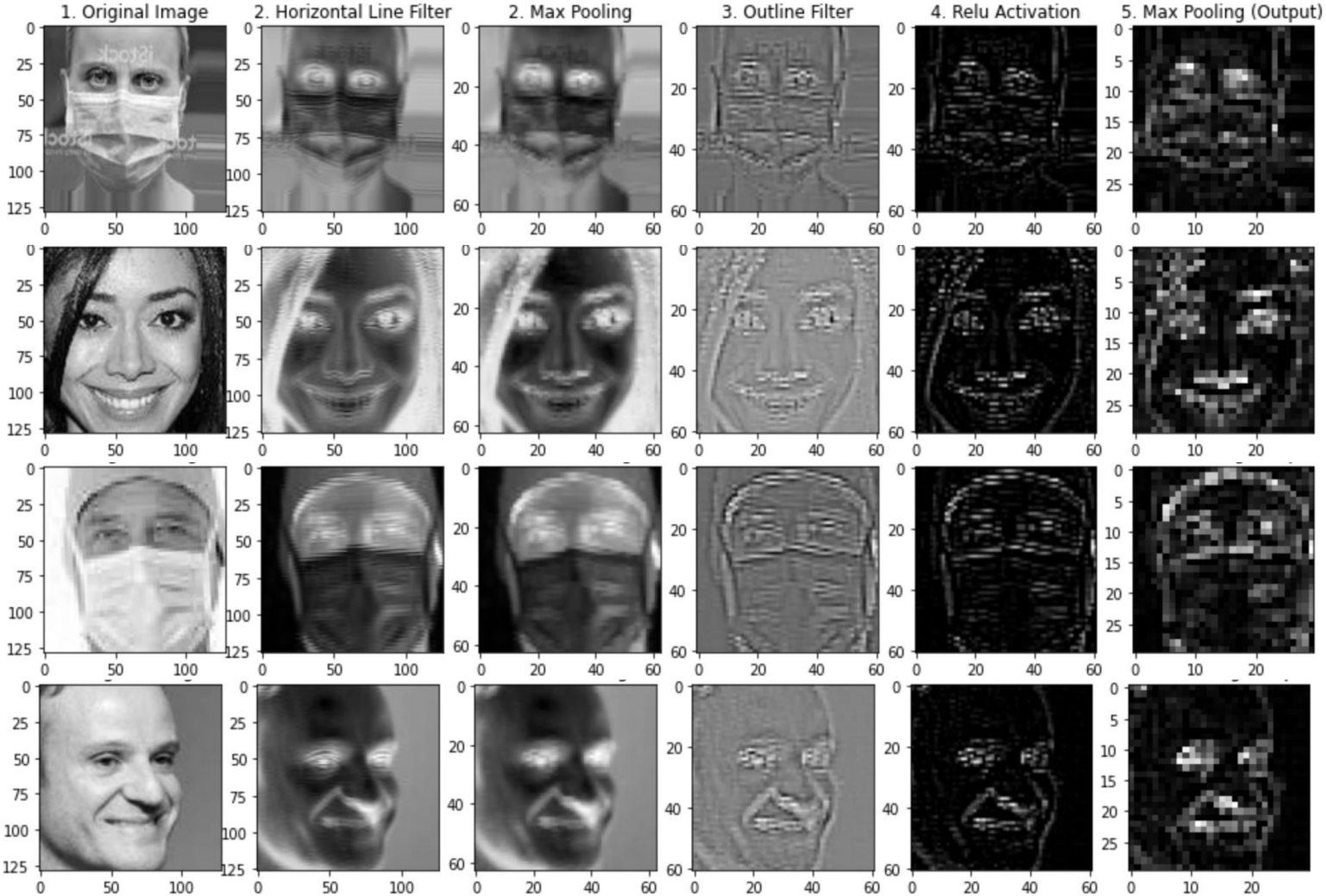
Pooling


Single depth slice
x
y
1 1 2 4
5 6 7 8
3 2 1 0
1 2 3 4
max pool with 2x2 filters
and stride 2
6 8
3 4

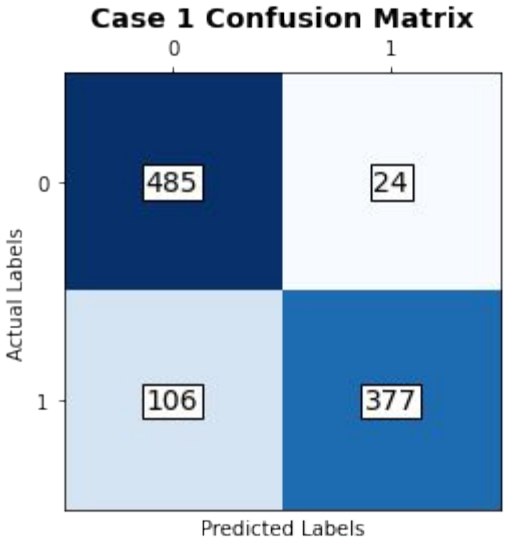
Relu Activation Function

$$f(u) = max(0, u)$$

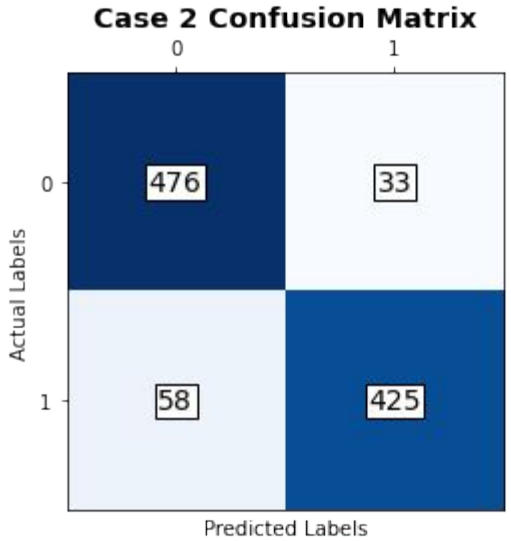



1. Original Image
2. Horizontal Line Filter
2. Max Pooling
3. Outline Filter
4. Relu Activation
5. Max Pooling (Output)

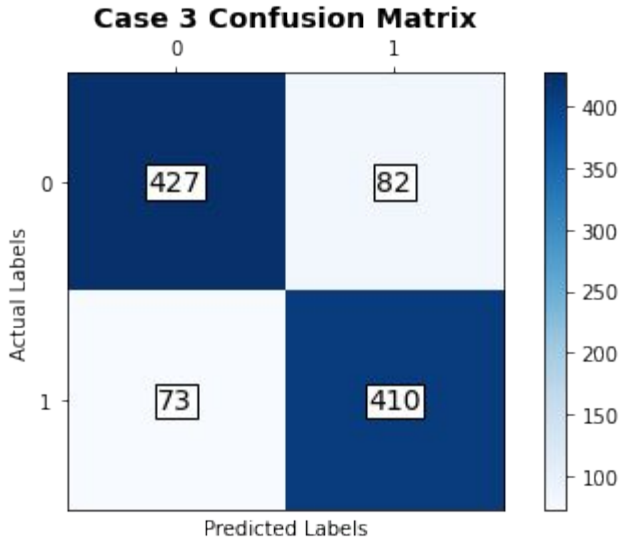
Classification – MPP



Overall: 0.8690
Class 0: 0.9528
Class 1: 0.7805



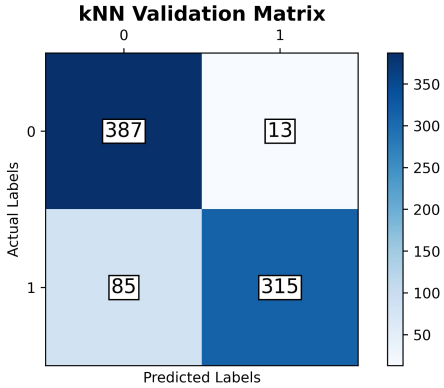
Overall: 0.9083
Class 0: 0.9352
Class 1: 0.8799

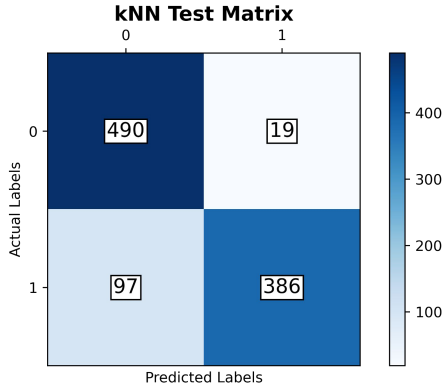


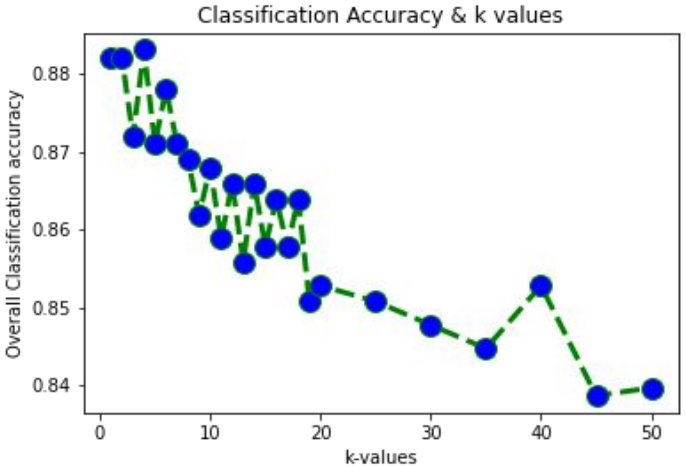
Overall: 0.8438
Class 0: 0.8389
Class 1: 0.8489

Classification – kNN

- k = 4 has the highest overall accuracy
- The larger k-value produces low accuracy



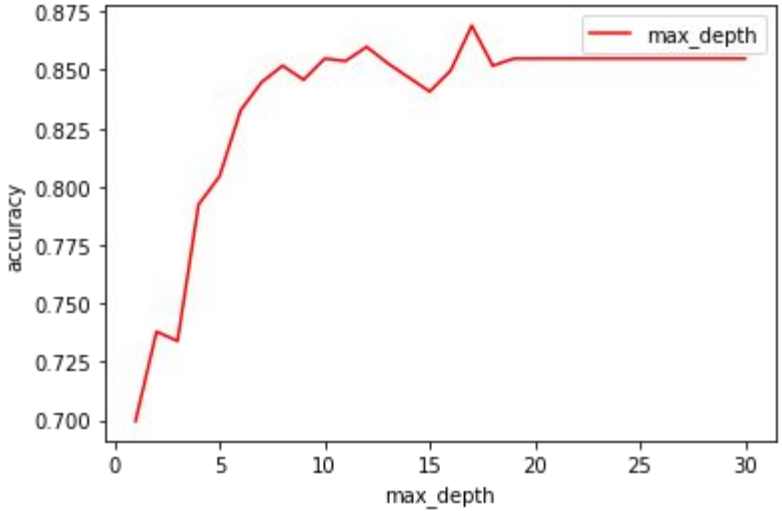




Overall: 0.8775
Class 0: 0.9675
Class 1: 0.7875

Overall: 0.88306
Class 0: 0.96267
Class 1: 0.79917

Classification – Decision Tree

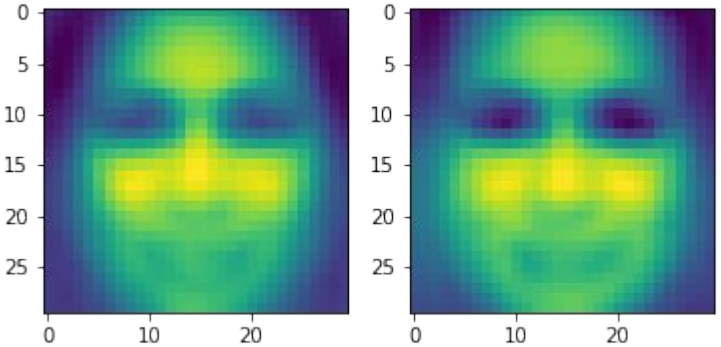


Fixing random_state=30, changing max_depth from 0 to 30, and find the maximum accuracy of 87% using sklearn.

Also, at the first stage we use a self-written decision tree code and use PCA to keep 100 dimension, but the accuracy is only 50%

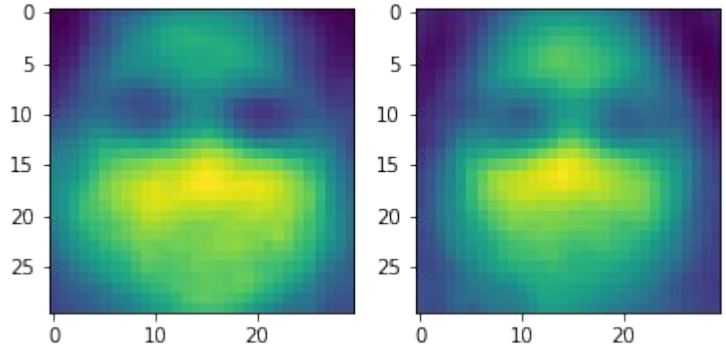
Classification – Clustering

kMeans



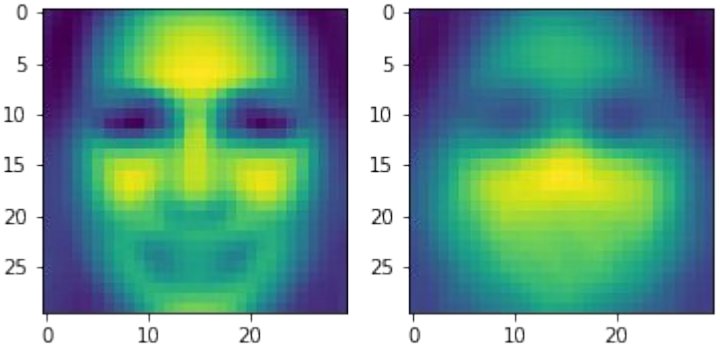
52.8%

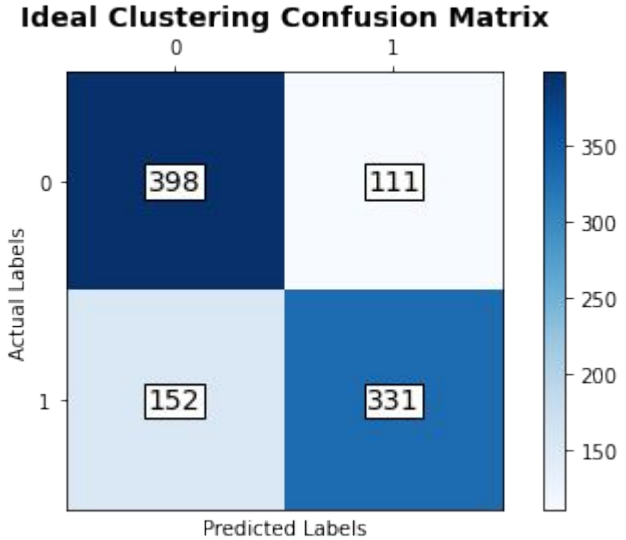
WTA



53.0%

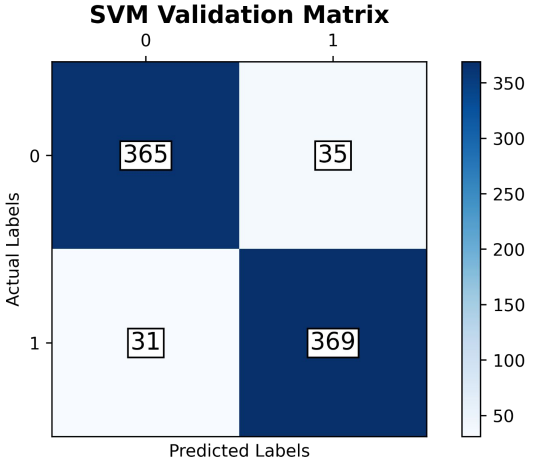
Classification – Clustering – Ideal



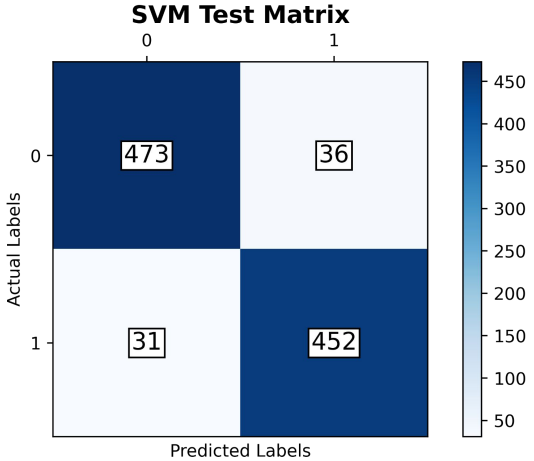


73.5%

Classification – SVM



Overall: 0.9175
Class 0: 0.9125
Class 1: 0.9225



Overall: 0.93245
Class 0: 0.92927
Class 1: 0.93581

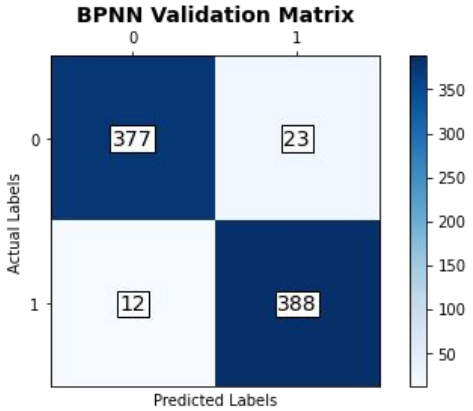
Classification – BPNN

Hyperparameter Tuning: Variation of ***Hyperband Tuning Method***

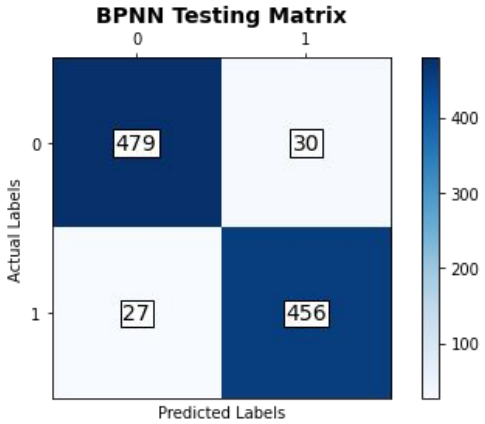
Best Hyperparameters:

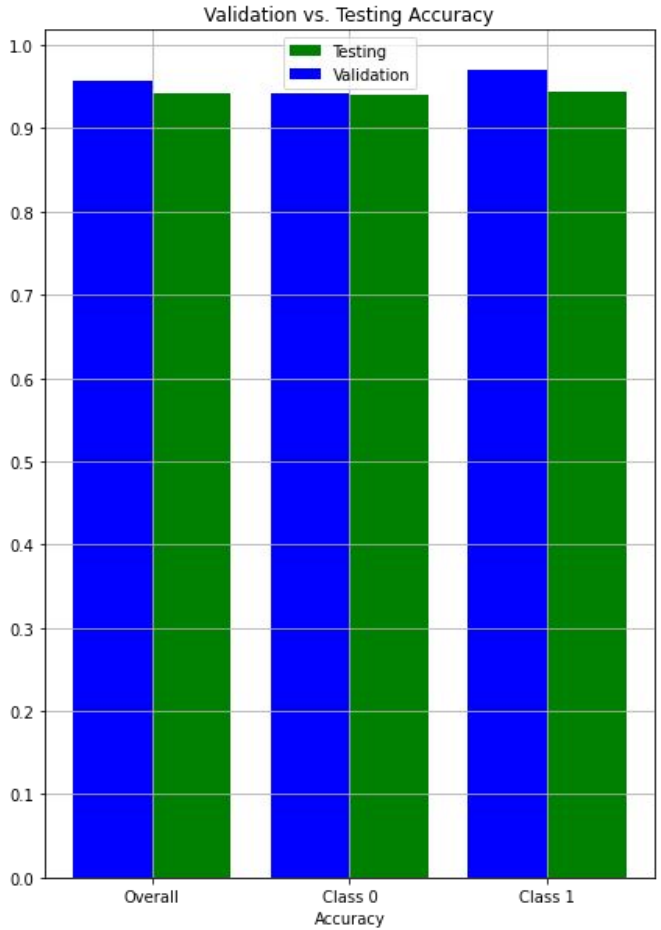
- Network Config = [900, 253, 2]
- MiniBatch Size = 1
- Learning Rate = 0.296296

Overall: 0.95625
Class 0: 0.9425
Class 1: 0.97

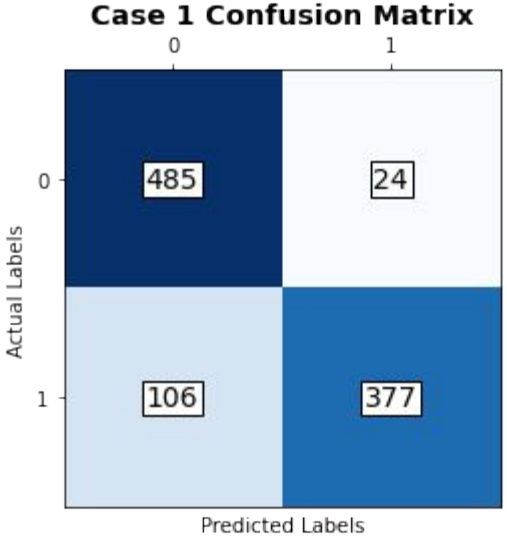


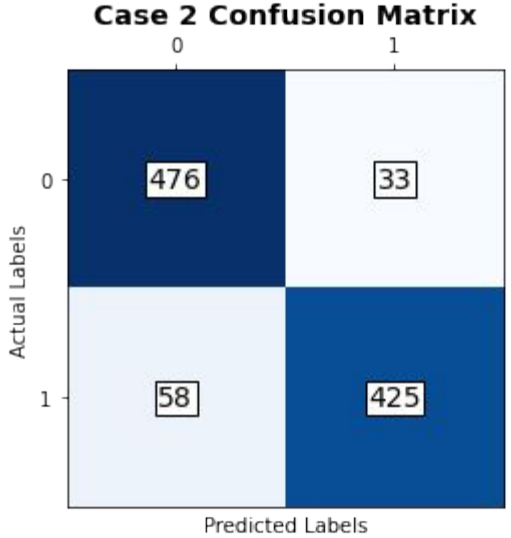
Overall: 0.94254
Class 0: 0.94106
Class 1: 0.94409

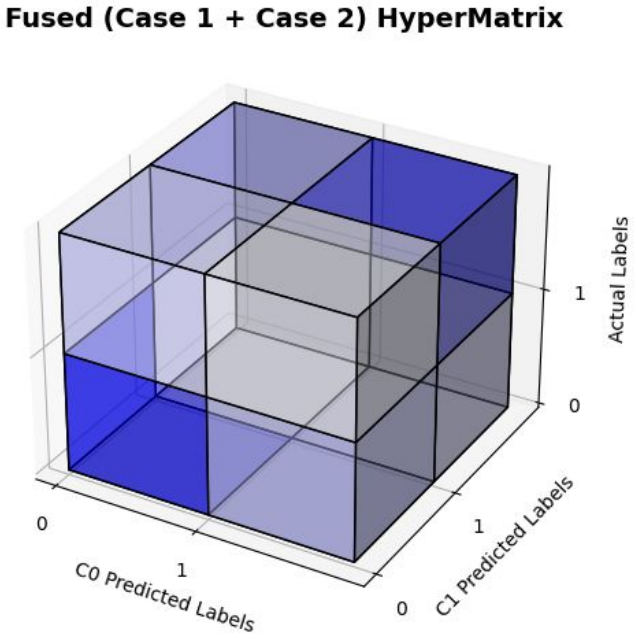




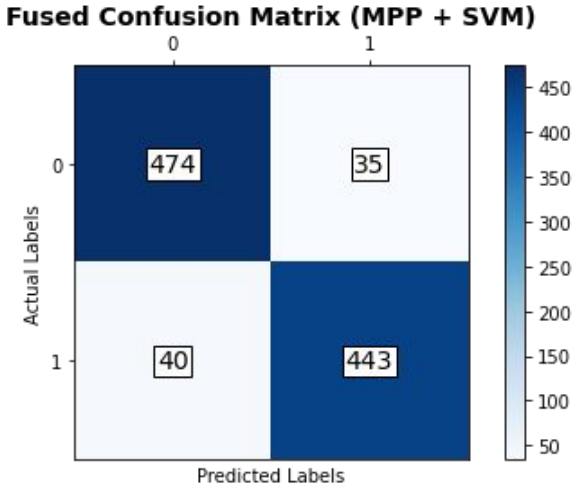
Classifier Fusion – Naive Bayes



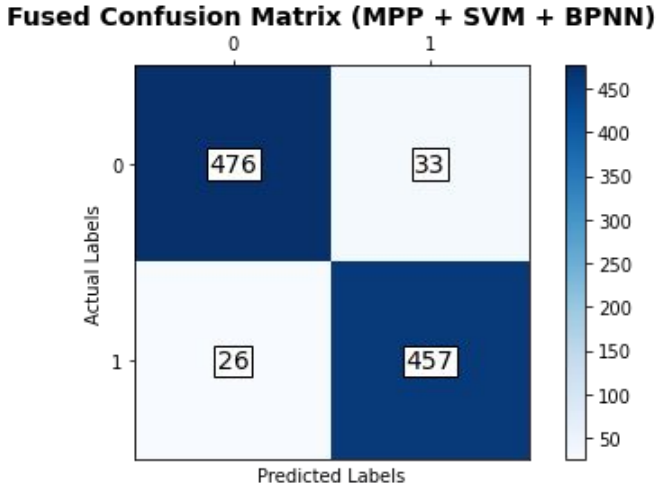




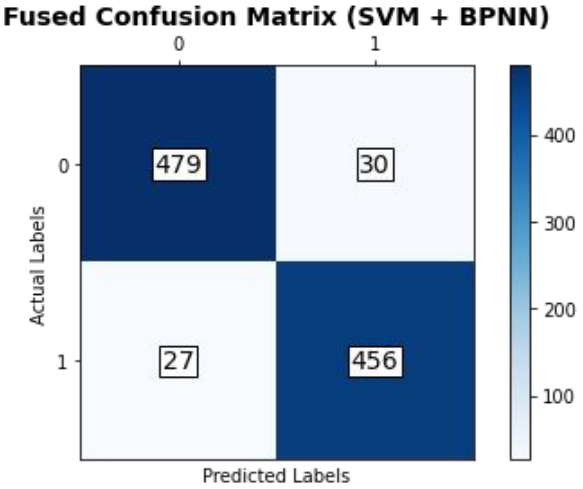
Classifier Fusion – Naive Bayes Continued



Overall: 0.92439
Class 0: 0.93123
Class 1: 0.91718



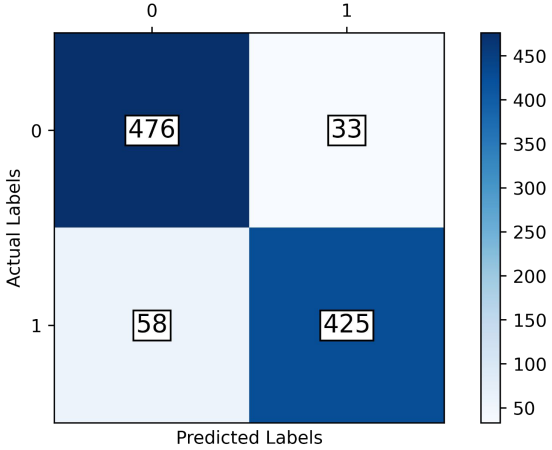
Overall: 0.94052
Class 0: 0.93516
Class 1: 0.94616



Overall: 0.94254
Class 0: 0.94106
Class 1: 0.94409

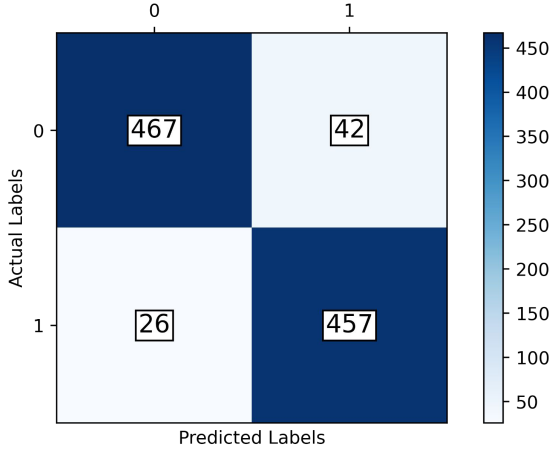
Classifier Fusion – BKS

BKS Confusion Matrix (Case 1,2,3)



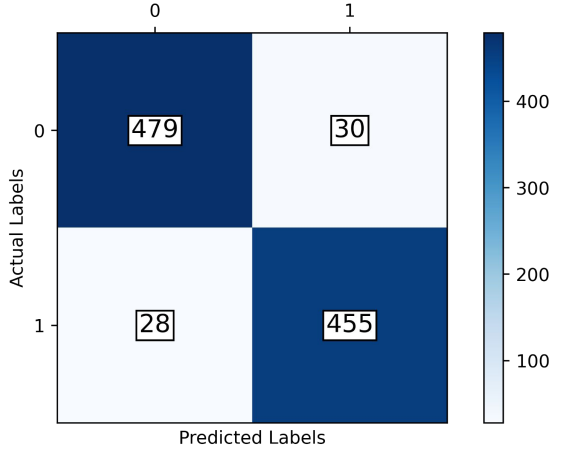
Overall: 0.90823
Class 0: 0.93516
Class 1: 0.87991

BKS Confusion Matrix (MPP + SVM)



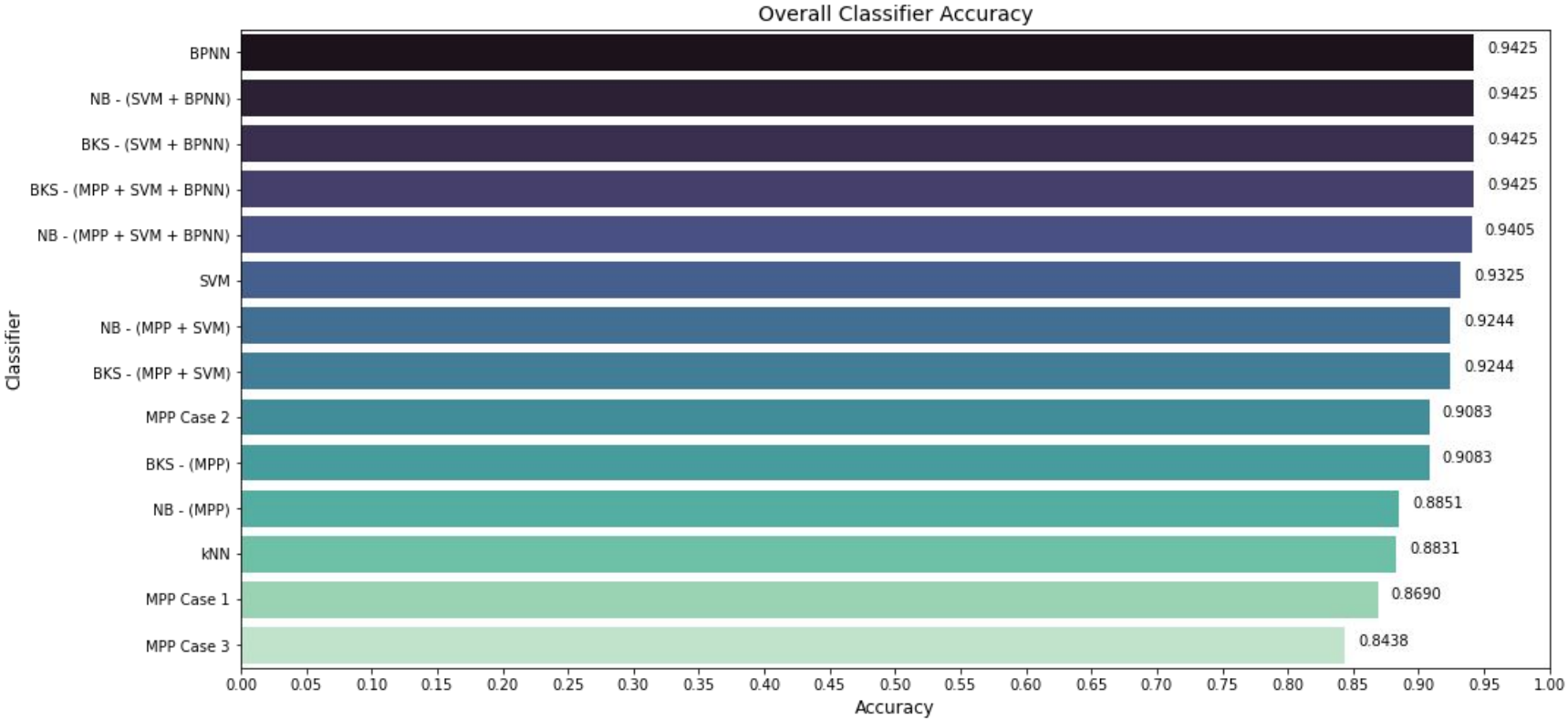
Overall: 0.93145
Class 0: 0.91748
Class 1: 0.94616

BKS Confusion Matrix (MPP + SVM + BPNN)



Overall: 0.94153
Class 0: 0.94106
Class 1: 0.94202

Evaluation


Overall Classifier Accuracy
Classifier
BPNN
NB - (SVM + BPNN)
BKS - (SVM + BPNN)
BKS - (MPP + SVM + BPNN)
NB - (MPP + SVM + BPNN)
SVM
NB - (MPP + SVM)
BKS - (MPP + SVM)
MPP Case 2
BKS - (MPP)
NB - (MPP)
kNN
MPP Case 1
MPP Case 3
0.9425
0.9425
0.9425
0.9425
0.9405
0.9325
0.9244
0.9244
0.9083
0.9083
0.8851
0.8831
0.8690
0.8438
0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70 0.75 0.80 0.85 0.90 0.95 1.00
Accuracy

Questions?