Classification Project

Nick Kozachuk, Carl Saba, Theodore Woodworth

Outline

➢ Data & Tasks
➢ Methodology
➢ Results

Data & Tasks

Data Sets Analyzed

- Iyer
- Cho
- YaleB

Cho Outline

- Collected from UCI ML repository
- Gene sequence data
- Sample Size: 386
- Total Features: 16
- Total Classes: 5

Iyer Outline

- Collected from UCI ML repository
- Gene sequence data
- Sample Size: 517
- Total Features: 12
- Total Classes: 11

YaleB Outline

- 3 Seperate sets of gray scale images of human faces from 38 people
- Sample Size:
  - Training: 2186
  - Testing: 228
- Total Features: 32x32
- Total Classes: 38

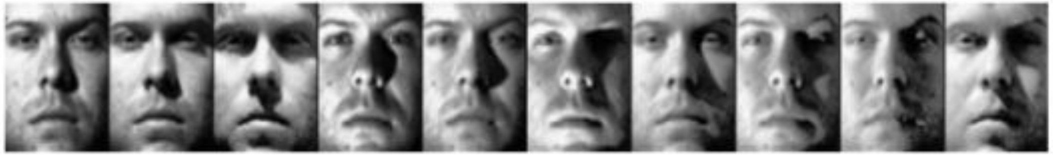


Figure 1: Sample images of one person in Yale B dataset.

Classification Methods

- Logistic Regression
- Random Forest
- Convolutional Neural Network (CNN)

Goal

- Analyze the performance of different classification methods across different datasets

Methodology

Libraries

- pROC
- Caret
- randomForest
- MLmetrics
- nnet
- keras
- tensorflow
- dplyr

Optimizations & Techniques

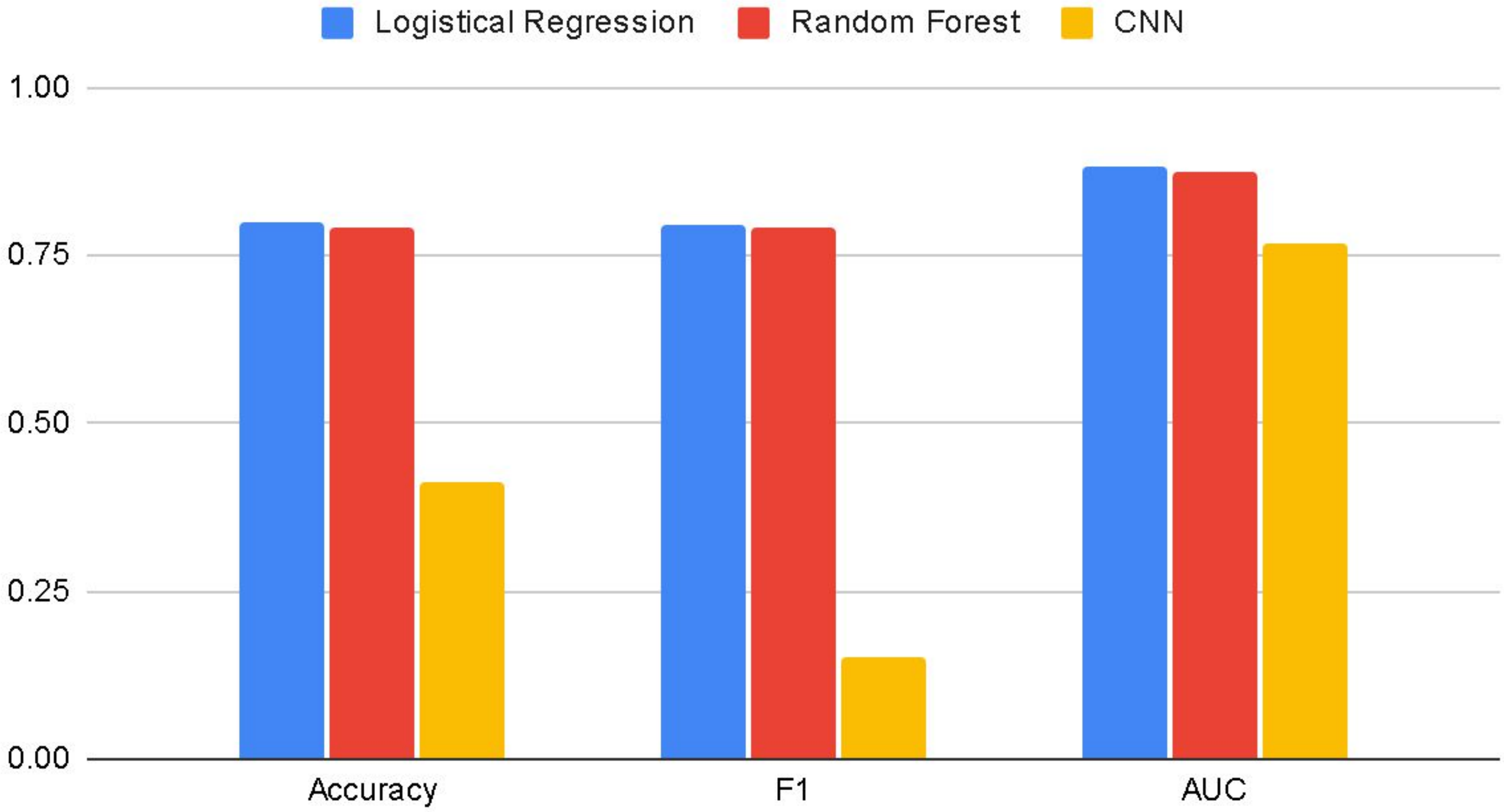
- K-fold Cross Validation
  - Iyer & Cho Hyperparameter tuning
- Principal Component Analysis (PCA)
  - Dimensionality Reduction
- Hold-out validation
  - YaleB provided a training & testing dataset
  - Iyer & Cho did not

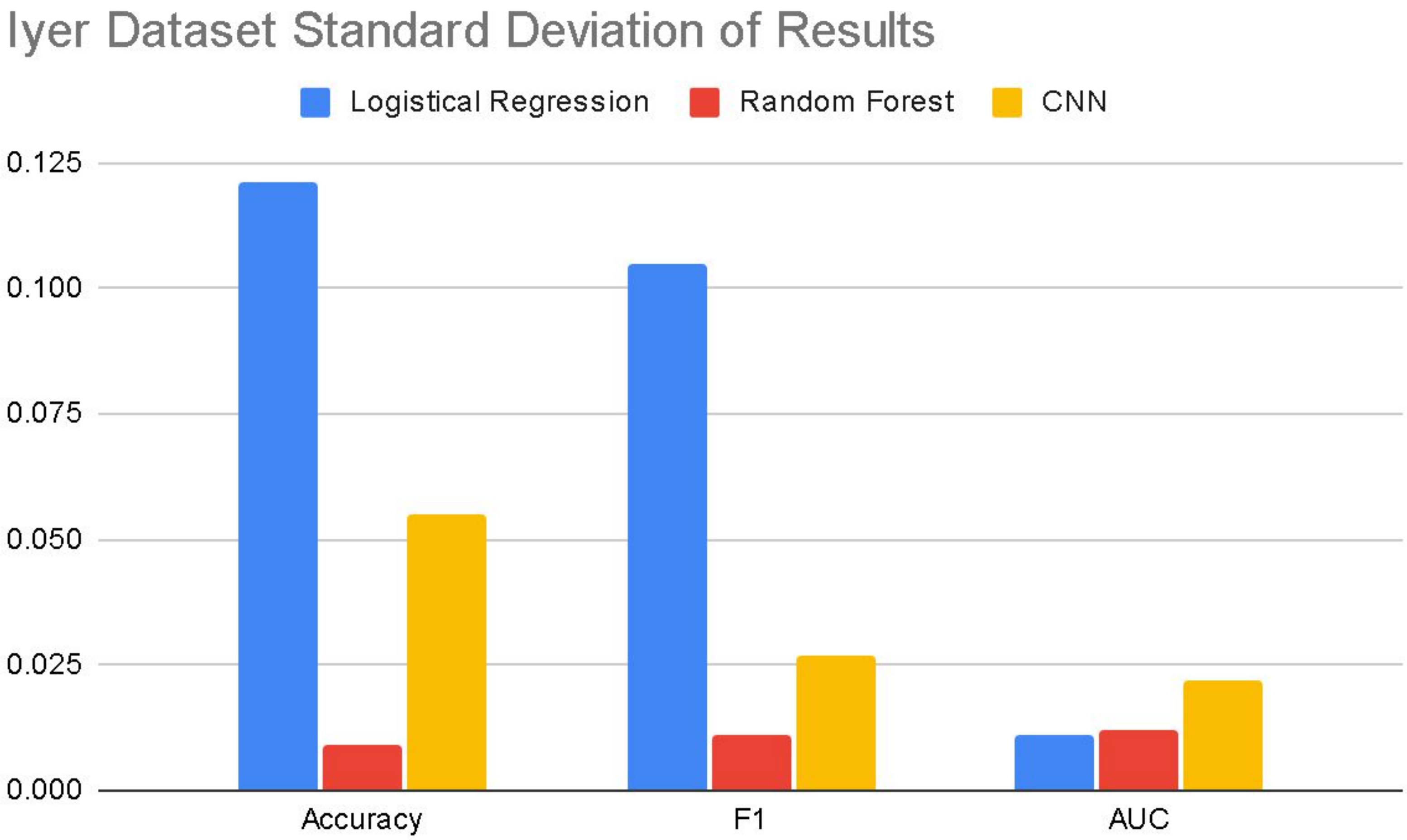
Results

Measurements

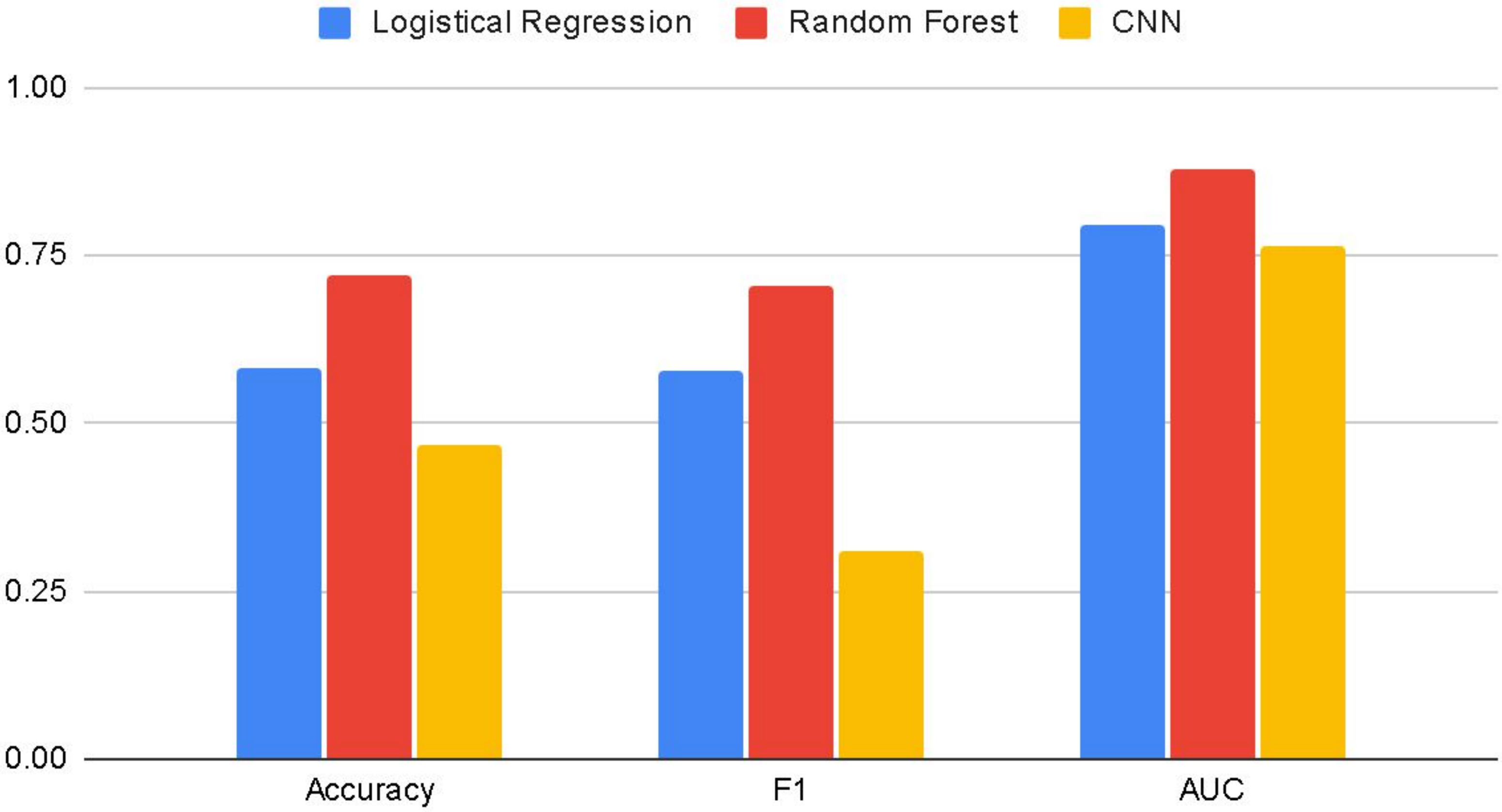
- F1 Score
- AUC
- Accuracy

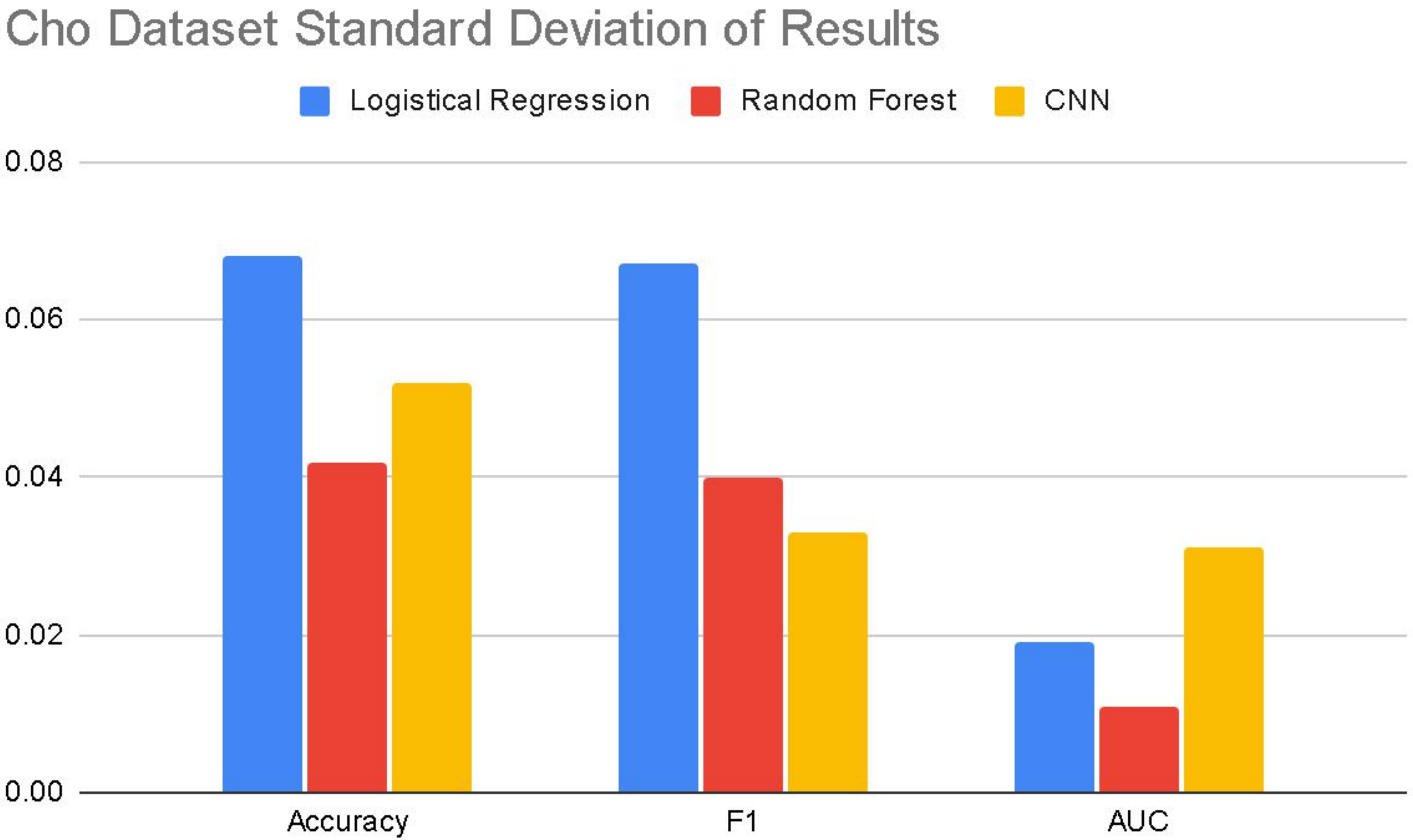
Iyer Dataset Results



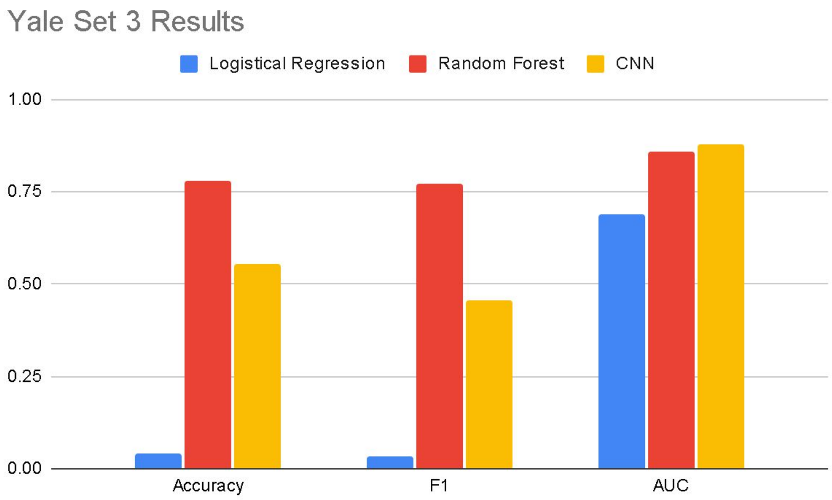

Iyer Dataset Standard Deviation of Results
Logistical Regression
Random Forest
CNN
0.125
0.100
0.075
0.050
0.025
0.000
Accuracy
F1
AUC

Cho Dataset Results

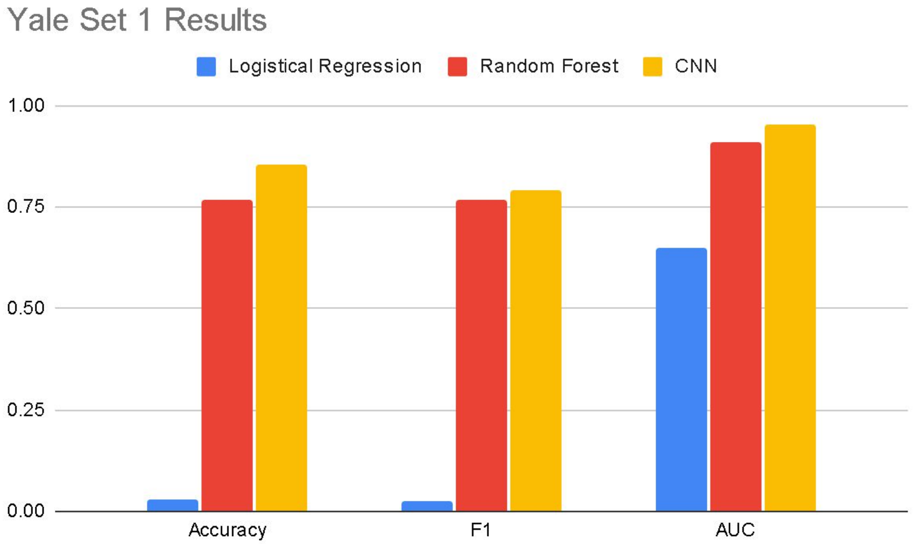




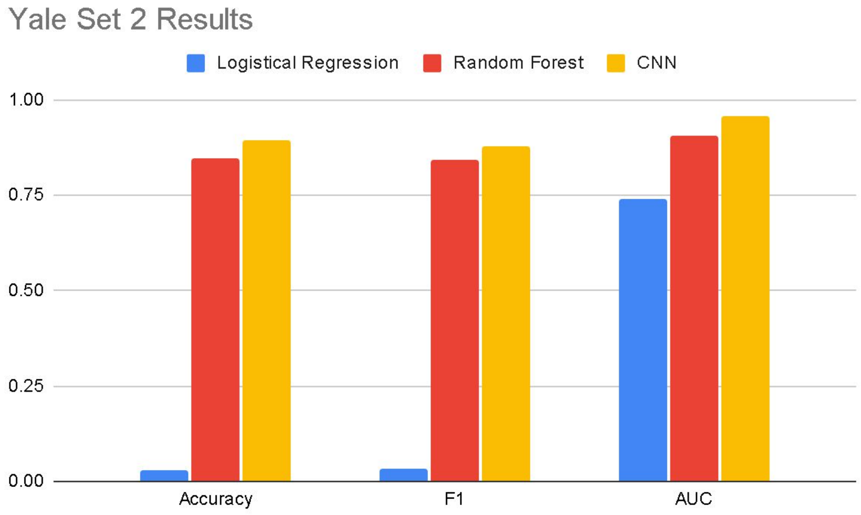
Cho Dataset Standard Deviation of Results



Yale Set 3 Results



Yale Set 1 Results



Yale Set 2 Results

Takeaways

Pros & Cons

- Logistic Regression
    - Simple, fast to train/test, less prone to overfitting
    - Struggles with high-dimensional data, sensitive to outliers, linear problems only, requires large datasets
- Random Forest
    - Handles large number of features, linear and nonlinear use, measures feature importance
    - Computationally intensive, suffers from overfitting if too many trees, difficult to interpret results
- Convolutional Neural Network
    - Handles high-dimensional data, captures nonlinear relationships, automatically learns features, powerful overall
    - Computationally intensive, requires accurate parameter specifications

Thank You