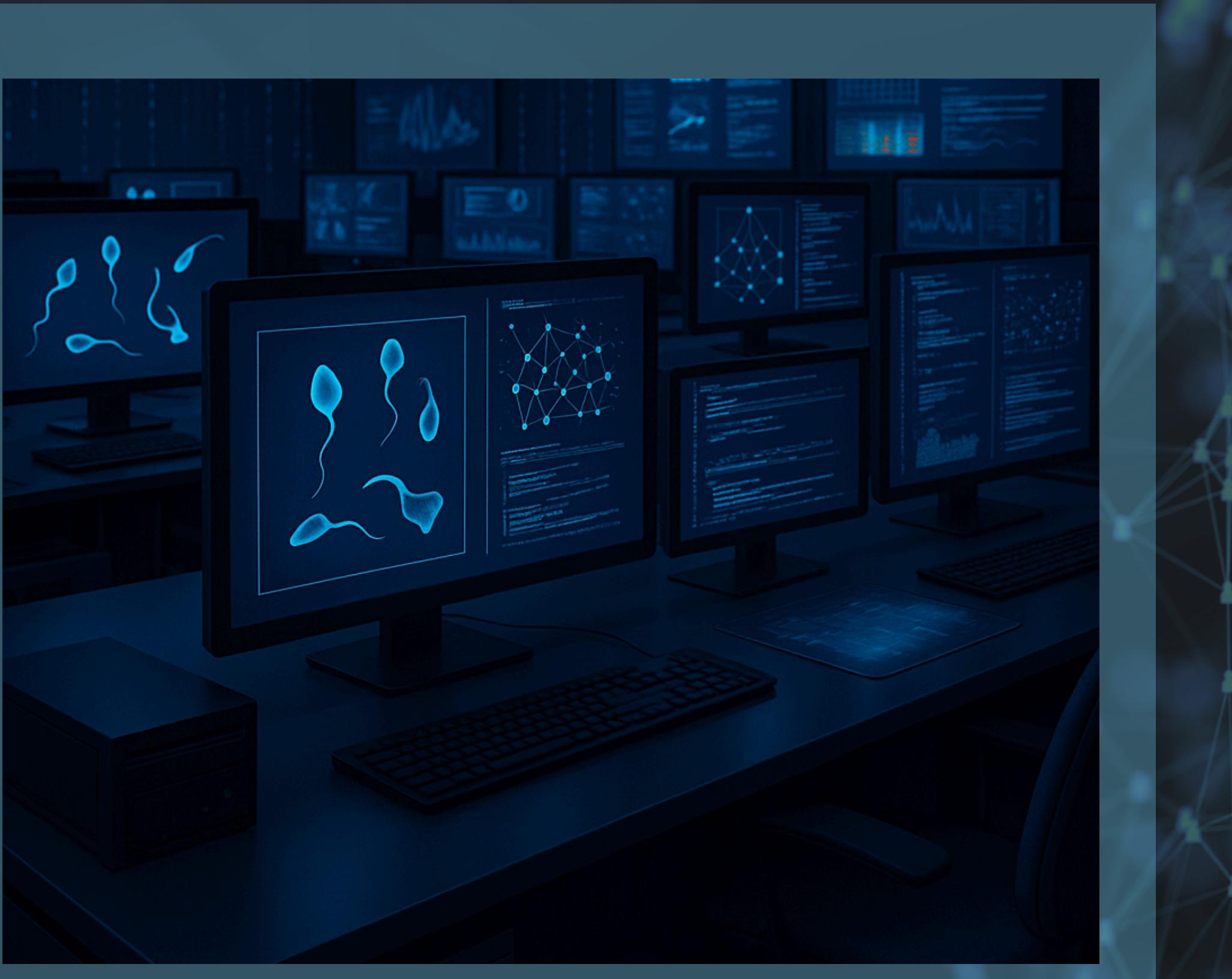


# MACHINE LEARNING BASED CLASSIFICATION OF SPERMS



**Artificial Intelligence  
Technische Hochschule  
Deggendorf**

Authors:

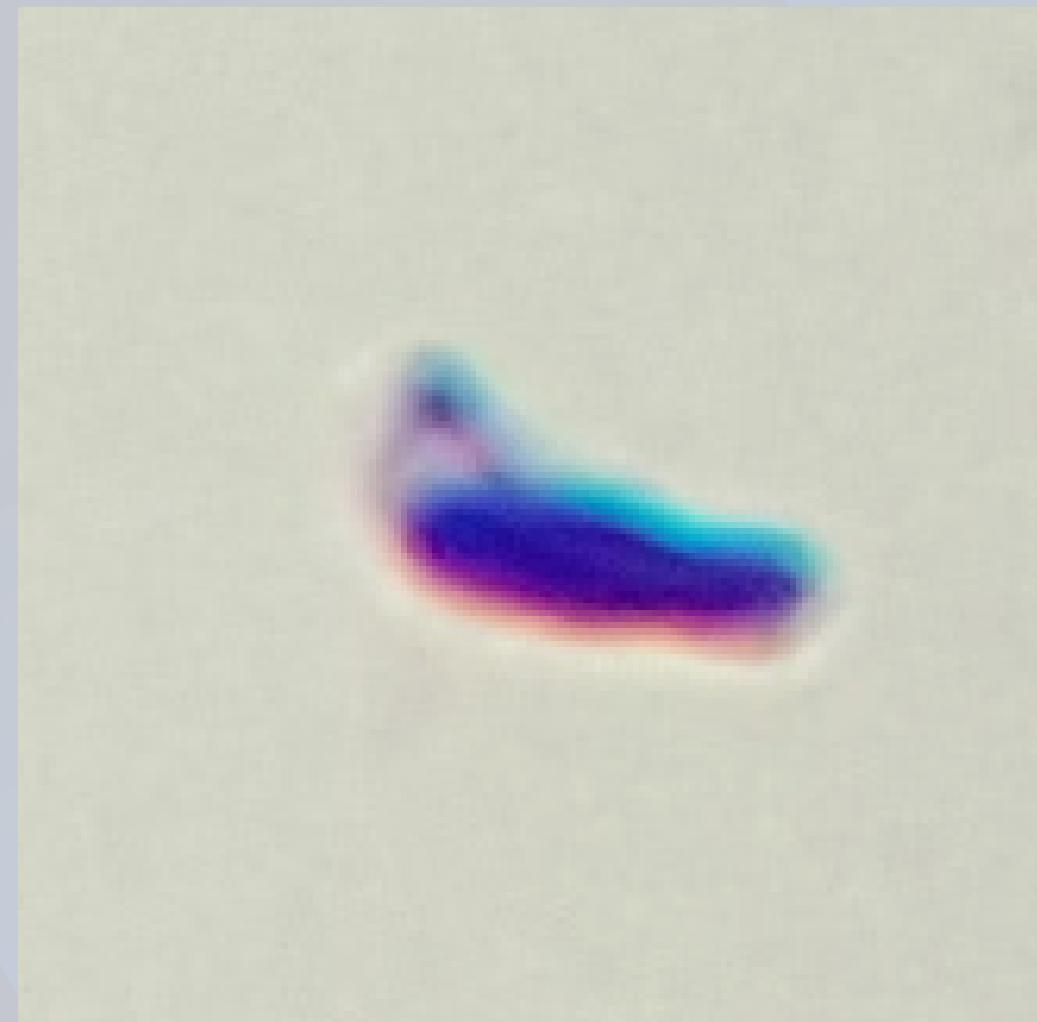
Aaditya Neupane  
Oween Barranzuela

# TOPIC OVERVIEW

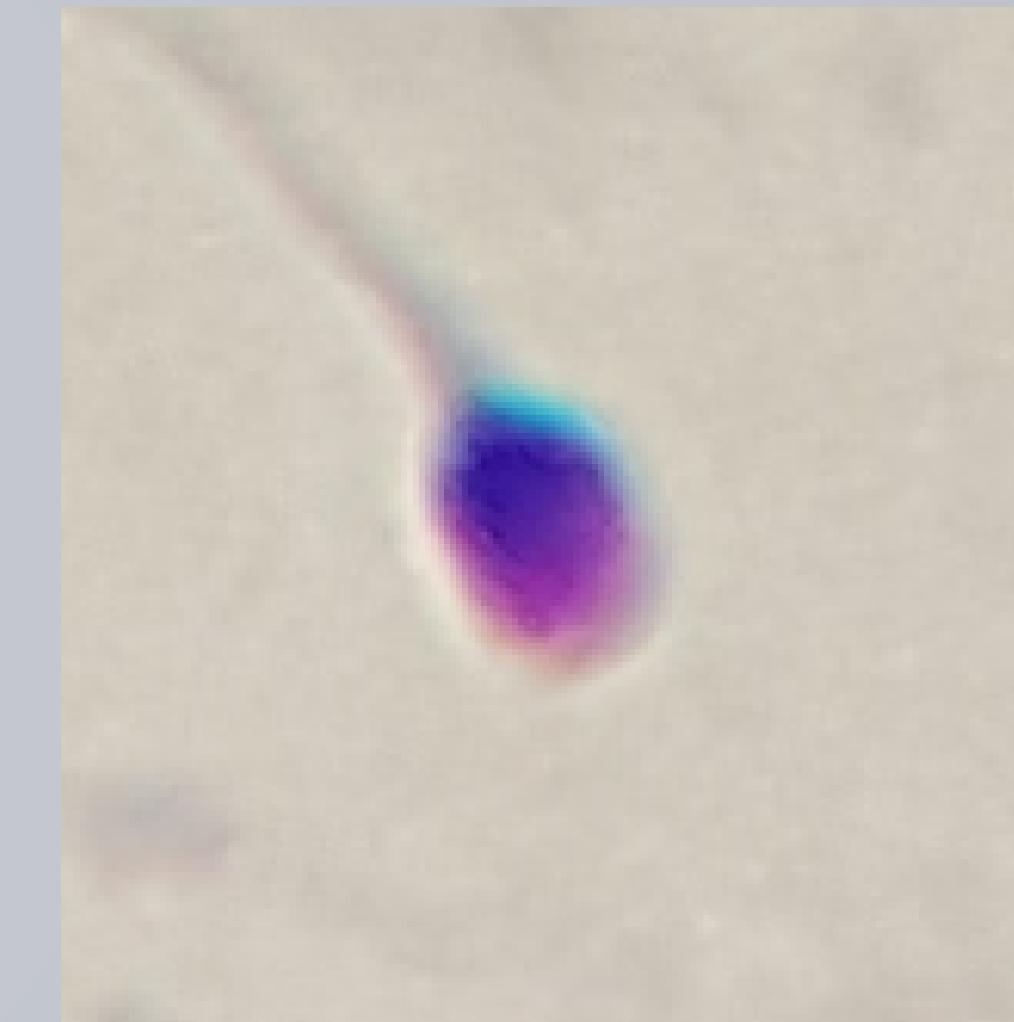
**Our project focuses on image-based classification of human sperm cells into three categories:**



**Abnormal Sperm**



**Non Sperm**



**Normal Sperm**

# OUR PURPOSE

## Research Contribution



- Most existing studies on sperm morphology focus solely on deep learning models, particularly Convolutional Neural Networks (CNNs).
- However, traditional machine learning approaches have not been explored or benchmarked on this dataset.

## Rise of AI in Healthcare



- Advances in machine learning and medical imaging are creating a growing demand for AI talent in healthcare.
- In this project we can gain practical experience in applying AI to real medical challenges

# Dataset Origin

## The Sperm Morphology Image Dataset (SMIDS)

Publicly available for free on Mendeley Data

Contributor:  
Hamza Ilham,  
Yildiz Technical  
University, Turkey

Created by capturing images using a smartphone attached to a microscope.

### First Application

This approach was originally introduced in a previous study for detecting and counting motile sperm cells.

### Further Applications

Applications of this dataset have mainly involved deep learning methods on platforms such as Kaggle.



# Our Methodology



**Data  
Exploration**

**Feature  
Extraction**

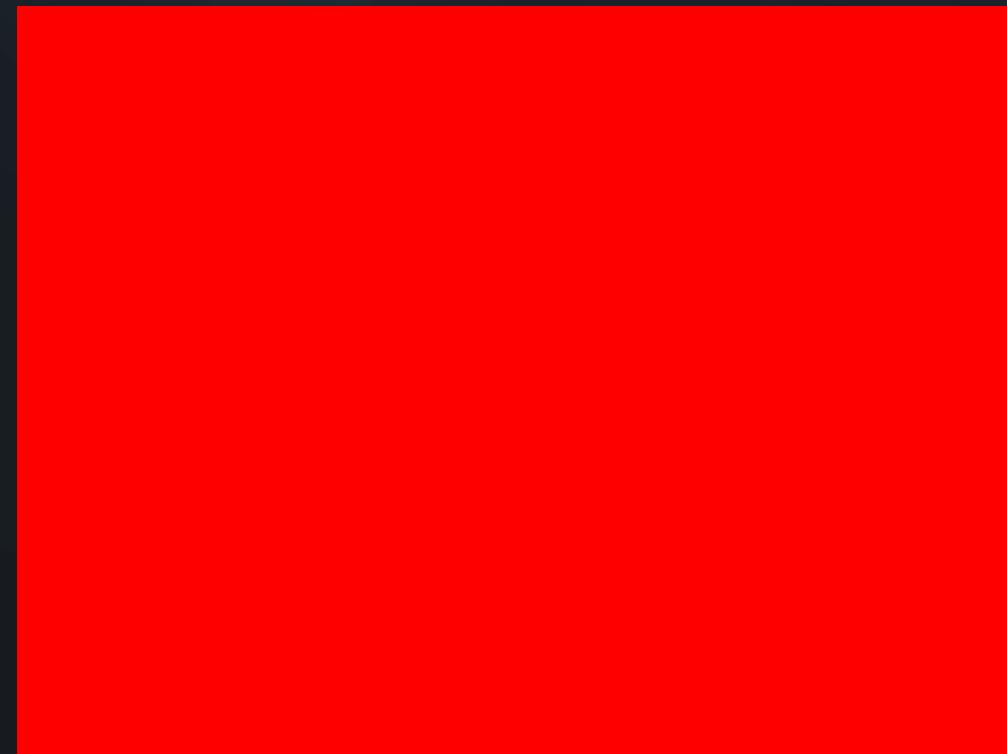
**Standardization**

**Model Training  
& Evaluation  
(GridSearch)**

# Data Exploration

# Color Channels

Red



Green

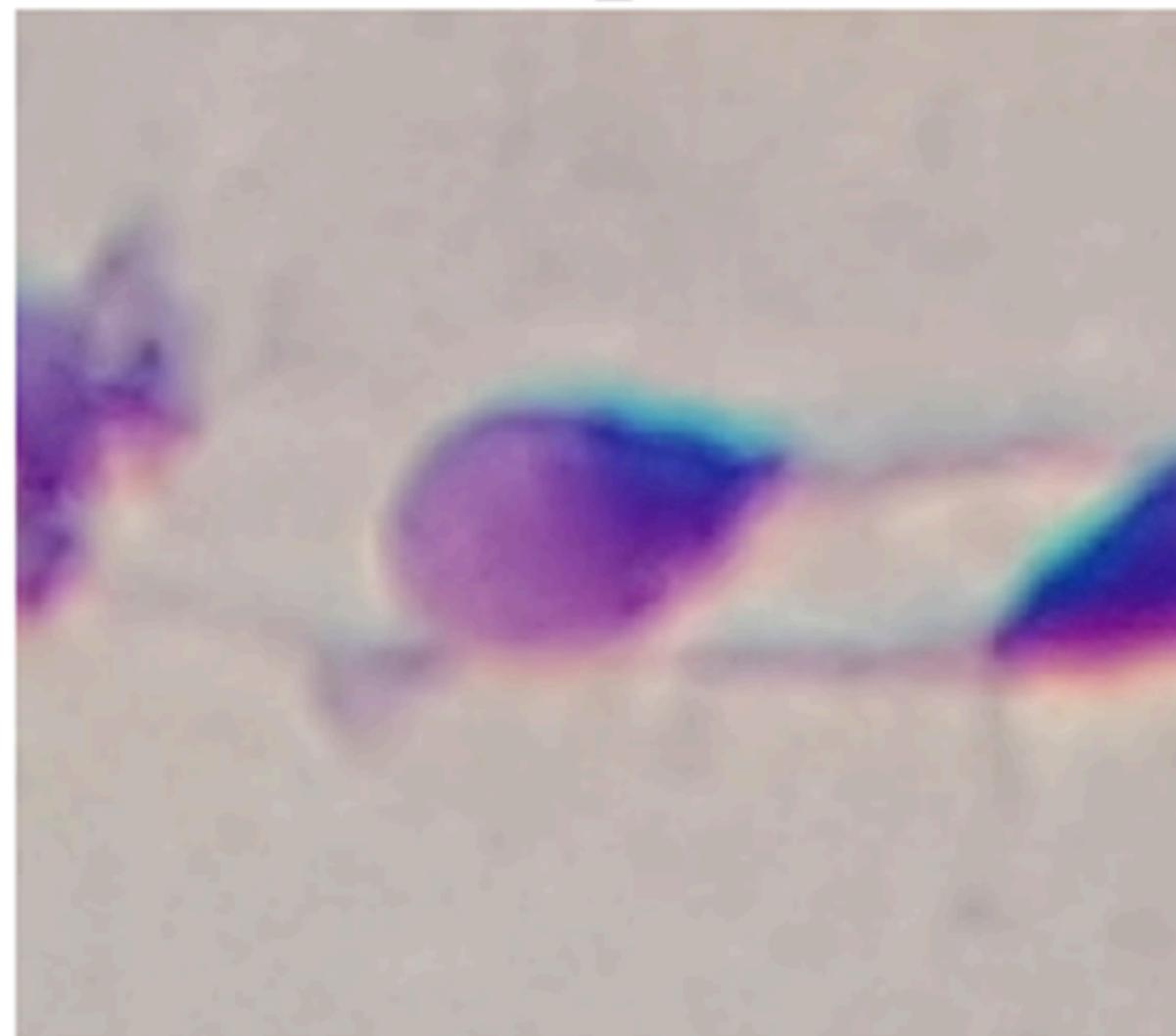


Blue

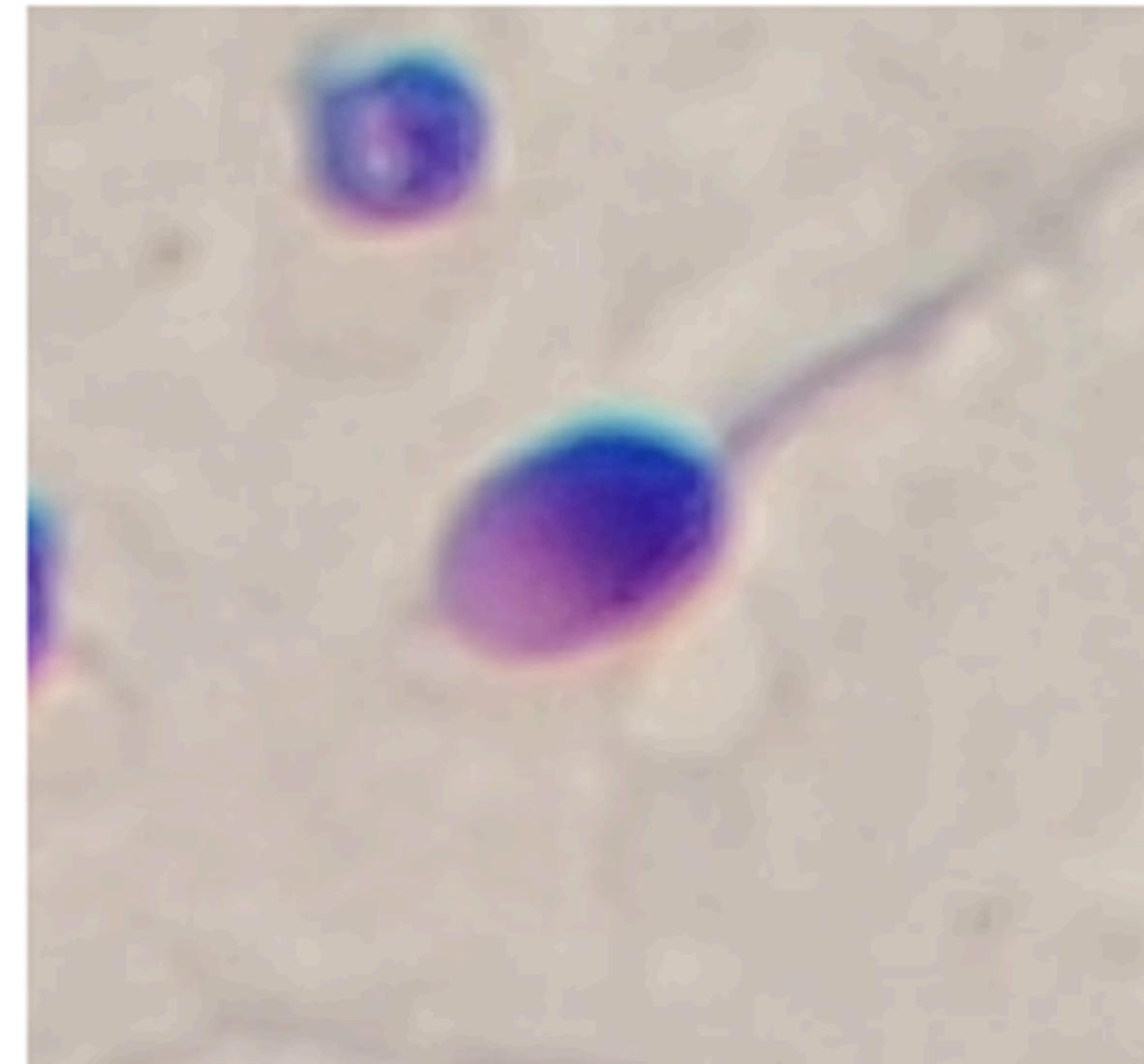


# Normal Sperm

Normal\_Sperm



Normal\_Sperm



Normal\_Sperm



# Abnormal Sperm

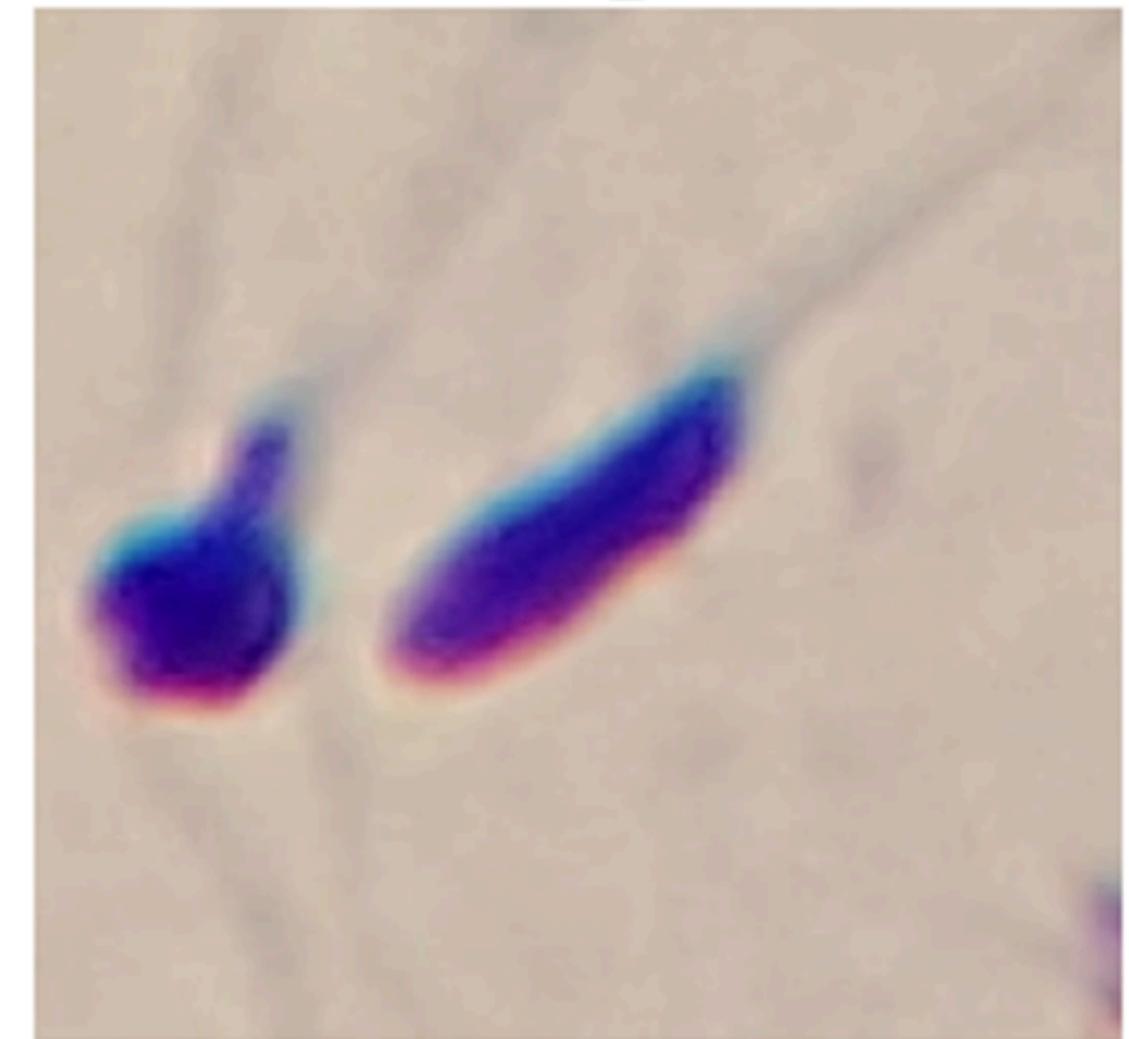
Abnormal\_Sperm



Abnormal\_Sperm

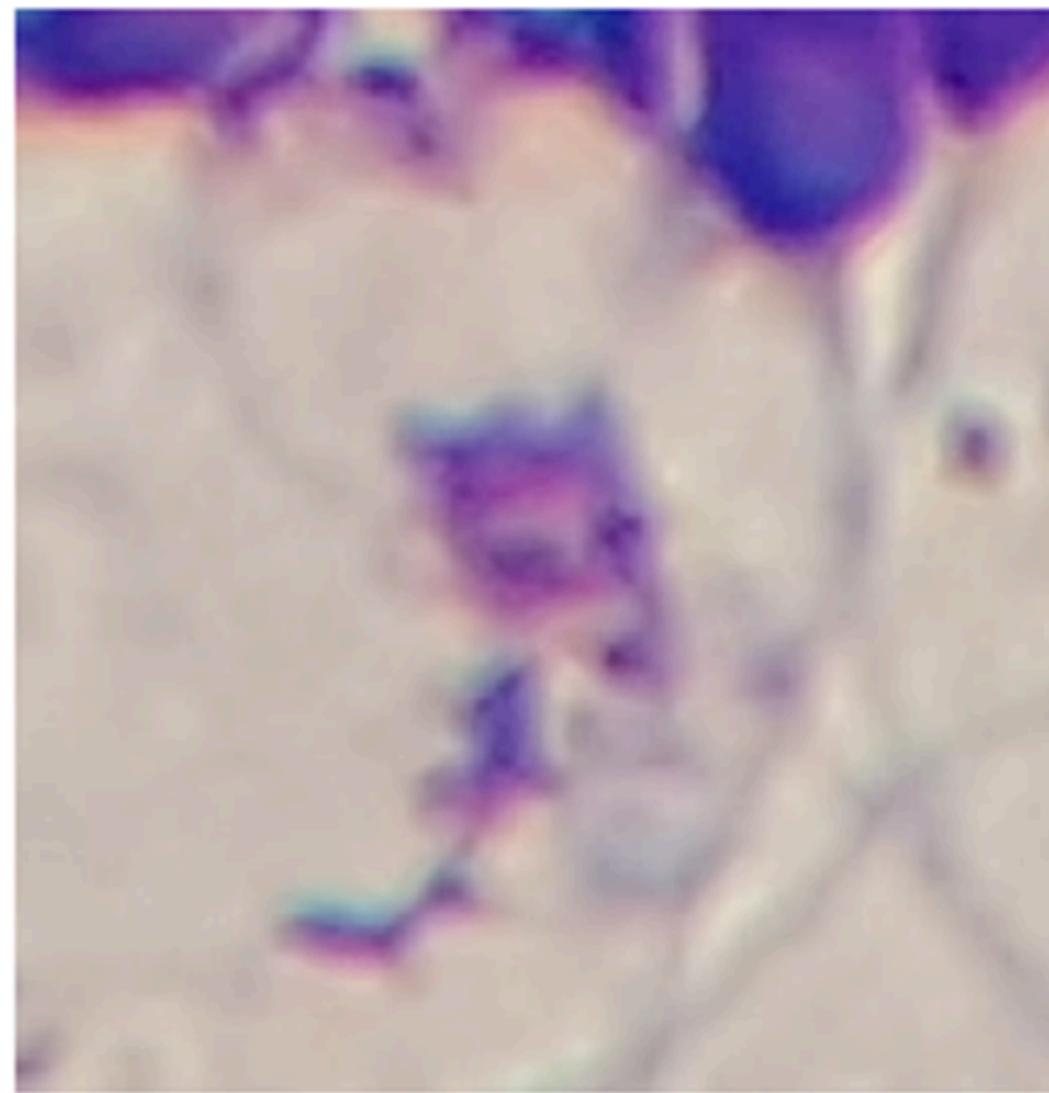


Abnormal\_Sperm

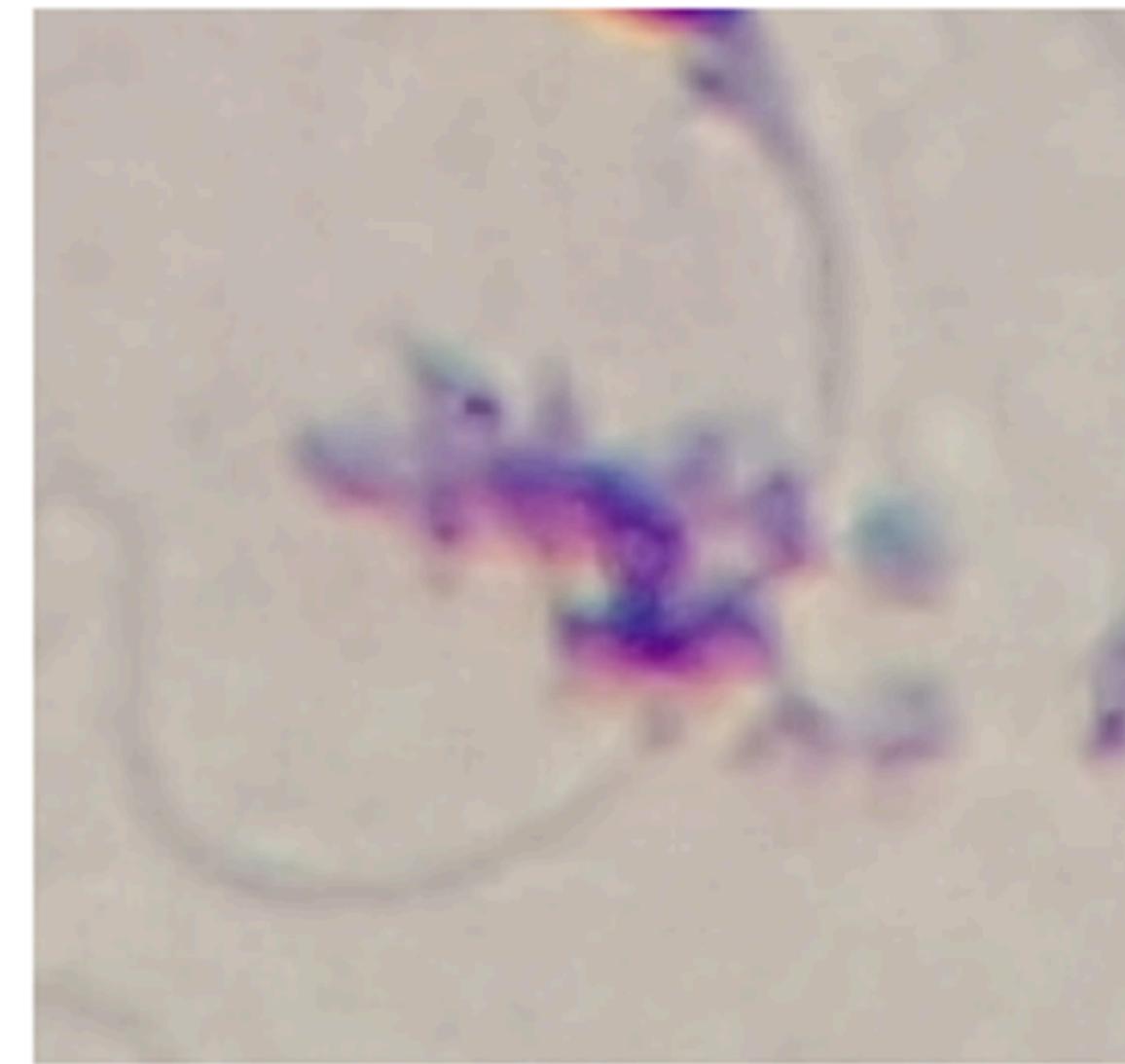


# Non Sperm

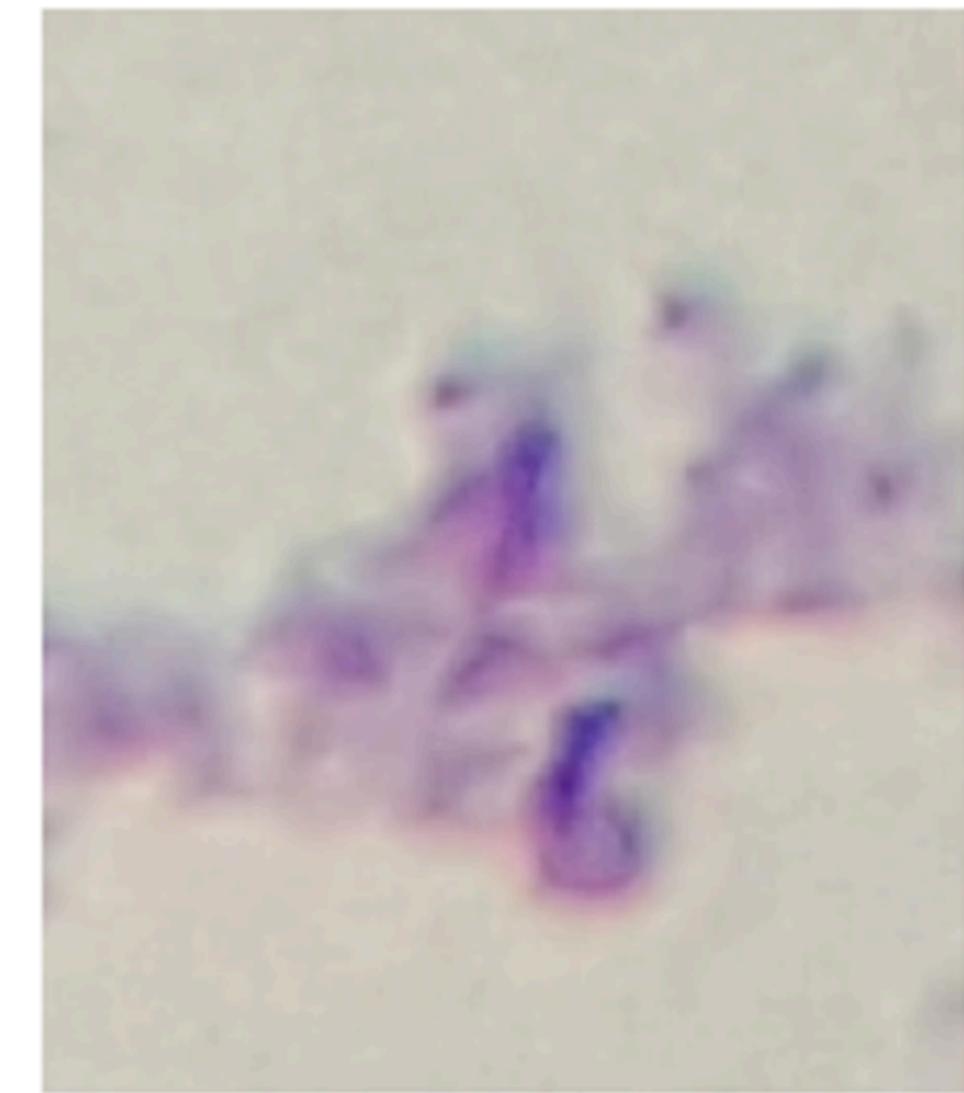
Non-Sperm



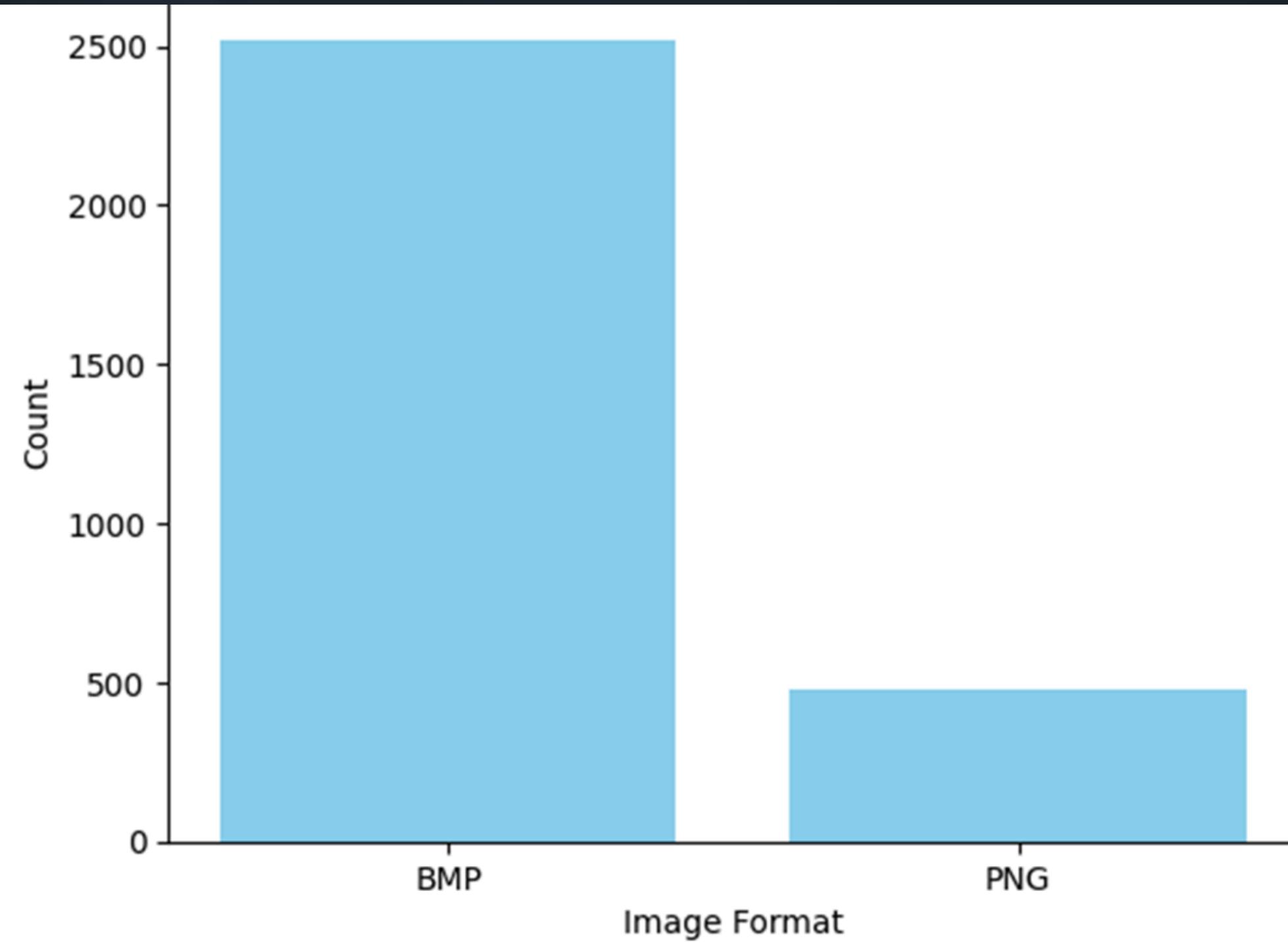
Non-Sperm



Non-Sperm



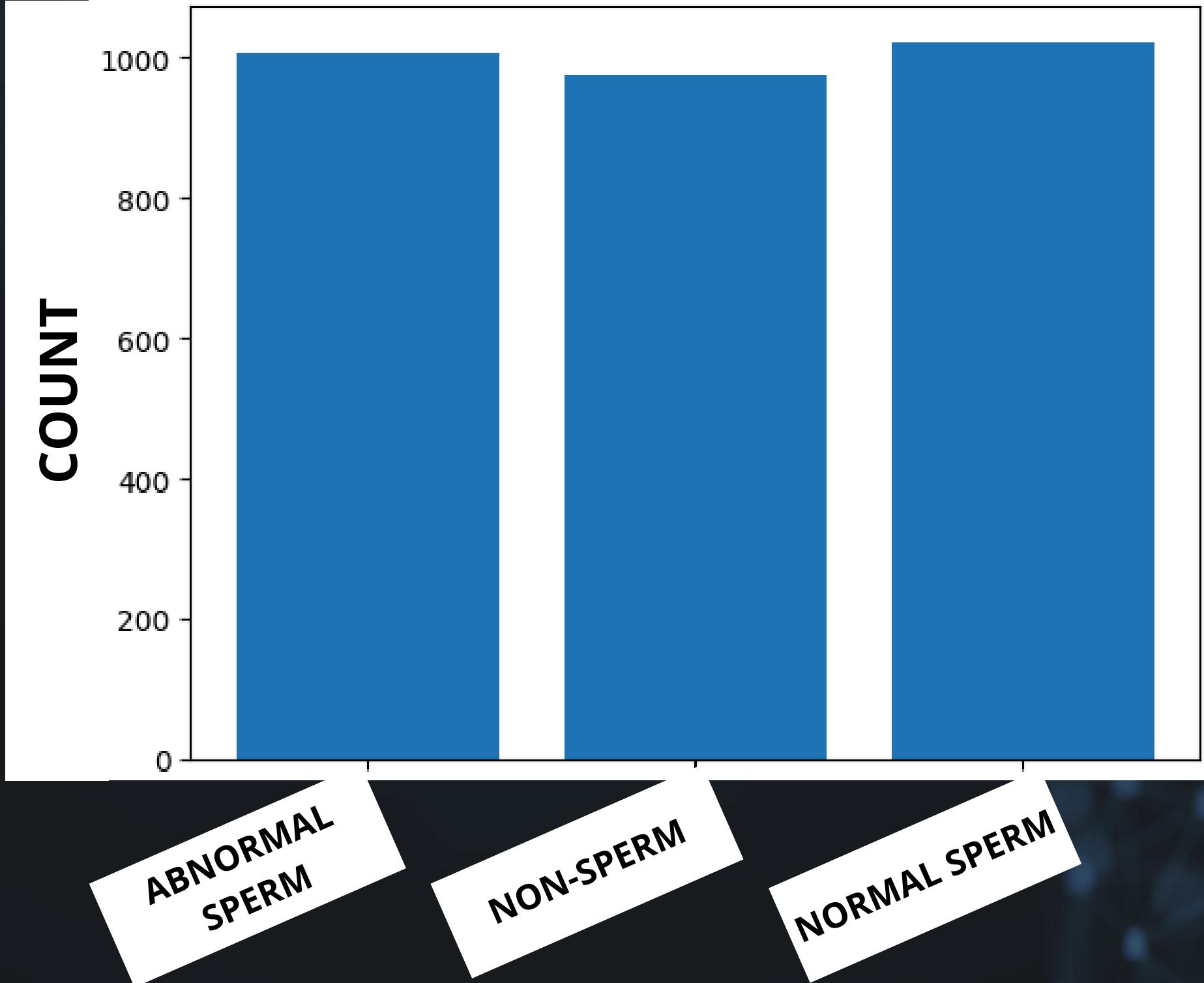
# Types of Image files



•BMP – 2520

•PNG – 480

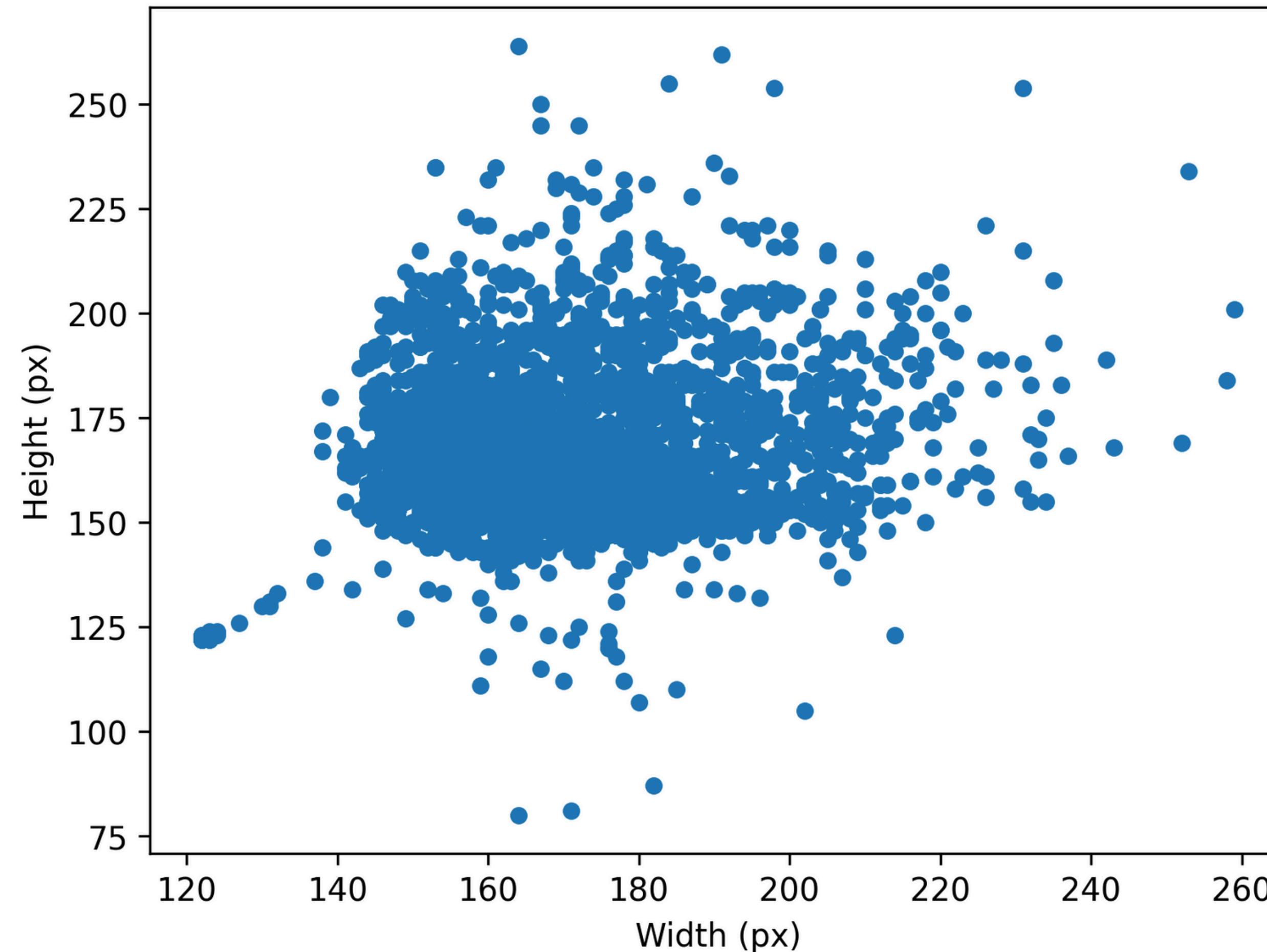
# Class distribution



**TOTAL IMAGES: 3000**

- Abnormal Sperm: 1005 images
- Non Sperm: 974 images
- Normal Sperm: 1021 images

# Scatter Plot of Image Sizes: Diversity Reveals Potential Issue



**Smallest image:**

**Width: 164 px**

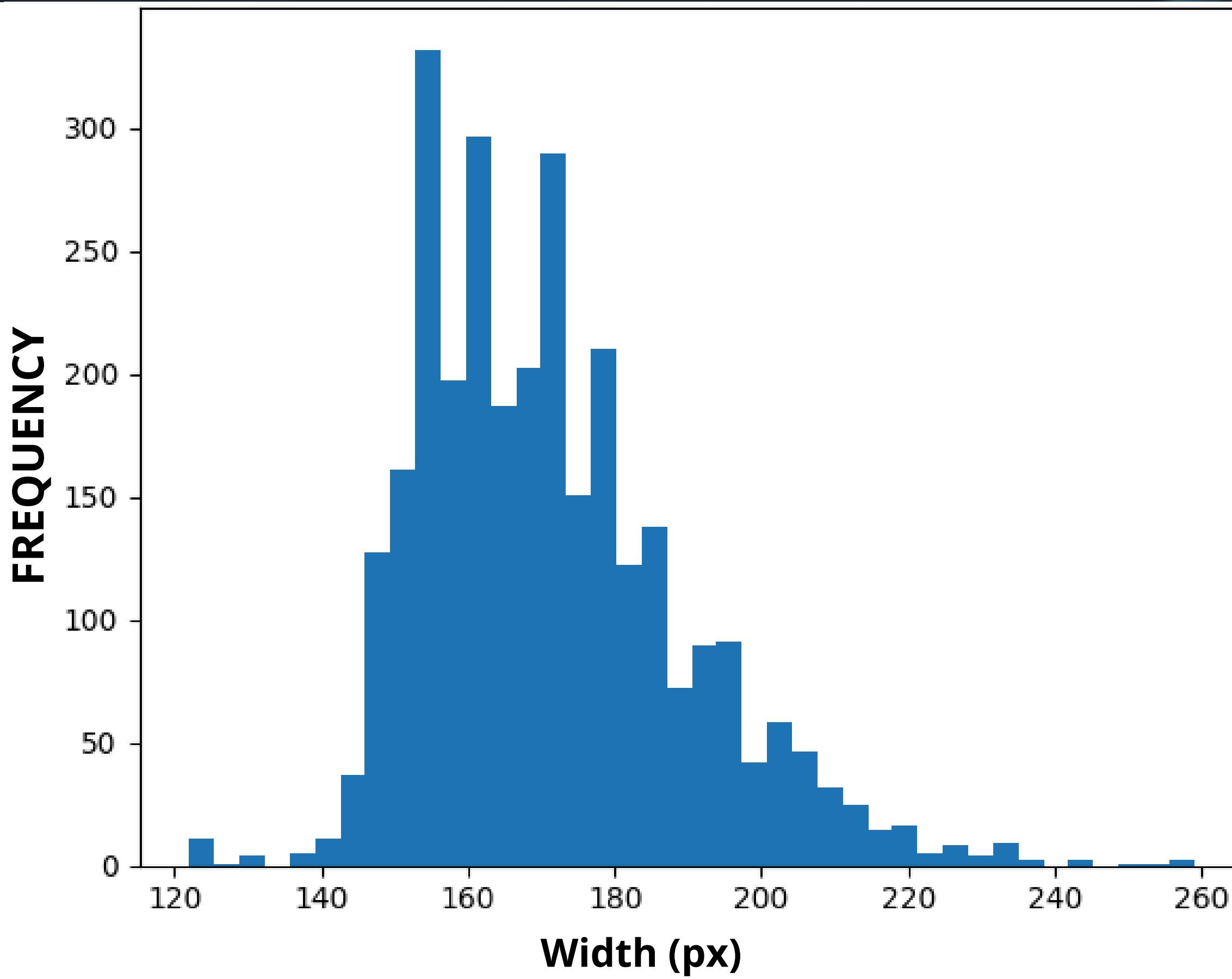
**Height : 80px**

**Largest Image:**

**Width : 253 px**

**Height: 234 px**

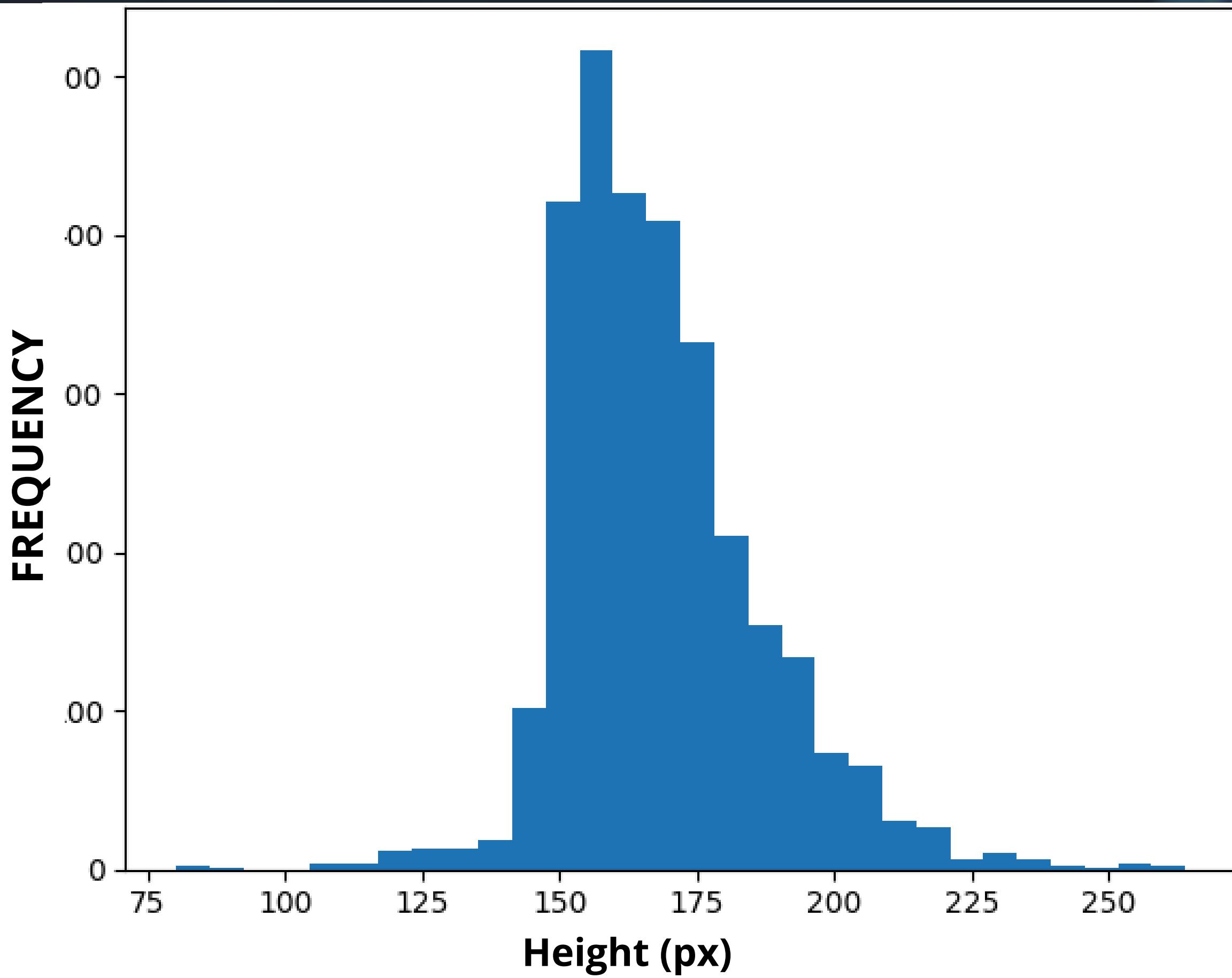
# IMAGE WIDTH DISTRIBUTION



**AVERAGE WIDTH :  
171.16 PX**

- **Minimum Width: 122 px**
- **Maximum Width: 259 px**

# IMAGE HEIGHT DISTRIBUTION



**AVERAGE HEIGHT :  
168.17 PX**

- Minimum Height: 80 px
- Maximum Height: 264 px



New shape :  
**(170,170)**

Question:  
**Why are we resizing?**

# Feature Extraction

# Features : two subgroups

## Color Channels Stats

- Mean
- Median
- Standard Deviation

- ⋮
- ## Color Histogram
- Intensity of each color channels across 8 bins

# How are we extracting the Basic Statistics?

2D ARRAY

What's really inside an image?

170				
123	123	...	123	255
145	100	...	56	156
...	...	...	...	...
145	45	...	56	221
100	56	...	23	15

Each color channel is just a grid of numbers.

$170 \times 170 = 28,900$  values per channel — the starting point for all our statistics.

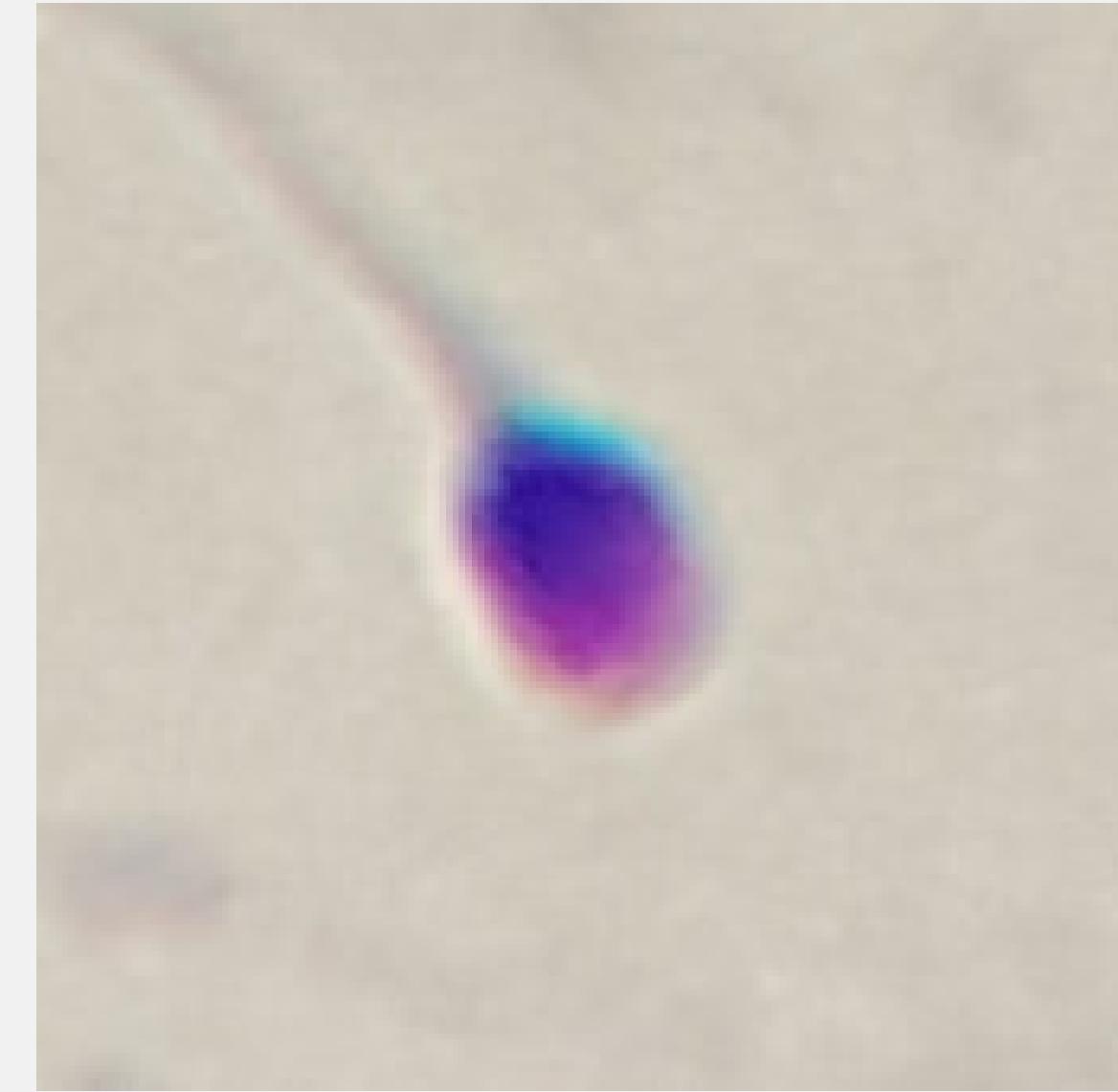
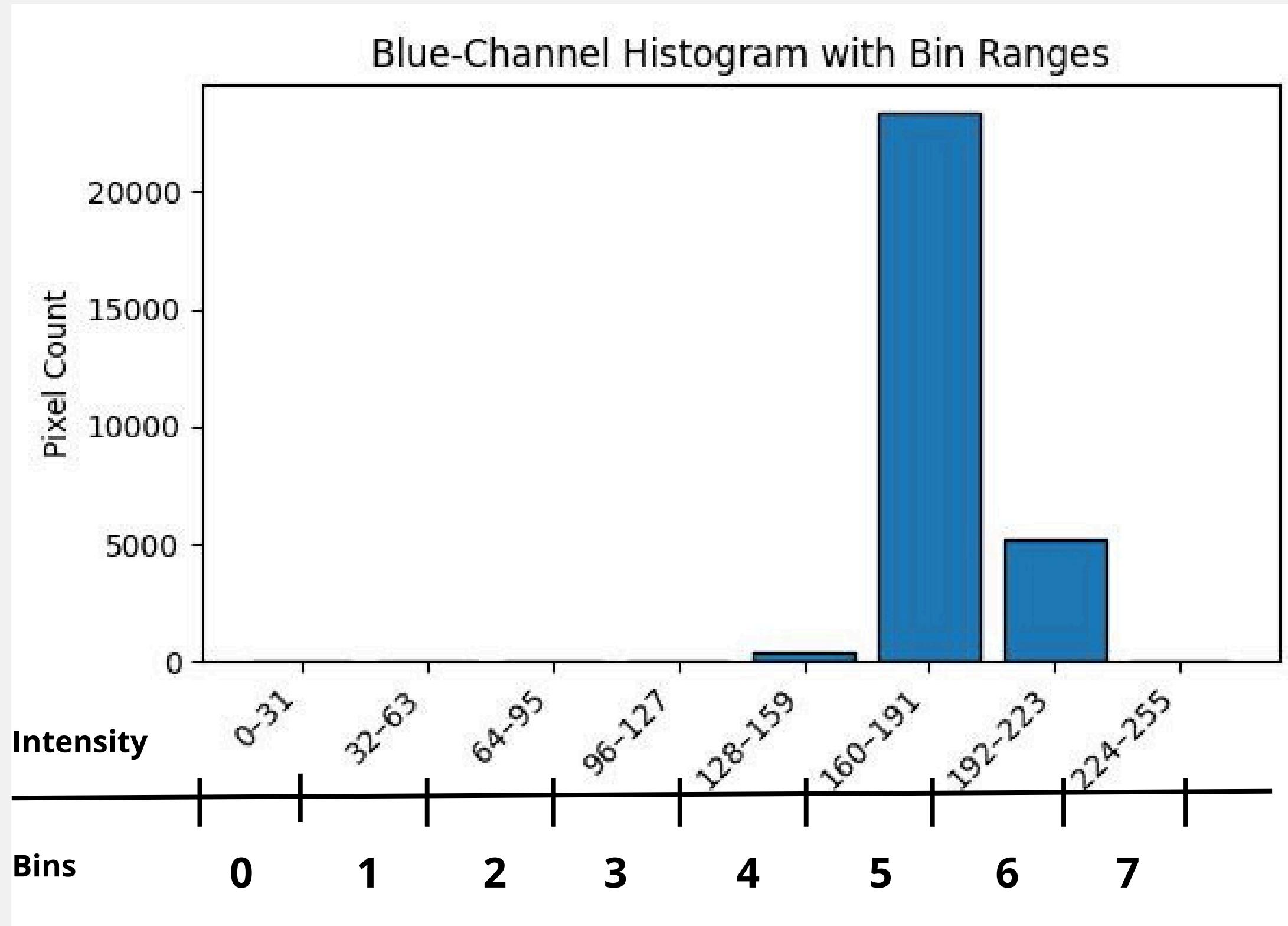
Treated as a 1D Vector .

We extract stats such as mean, std, and median.

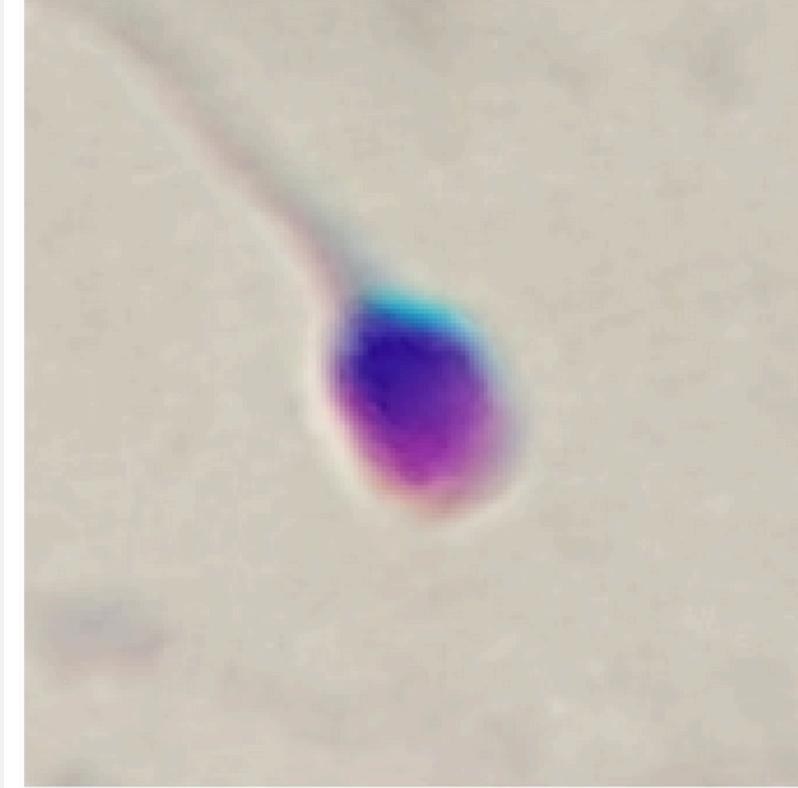
Why those values?

9 Features were extracted

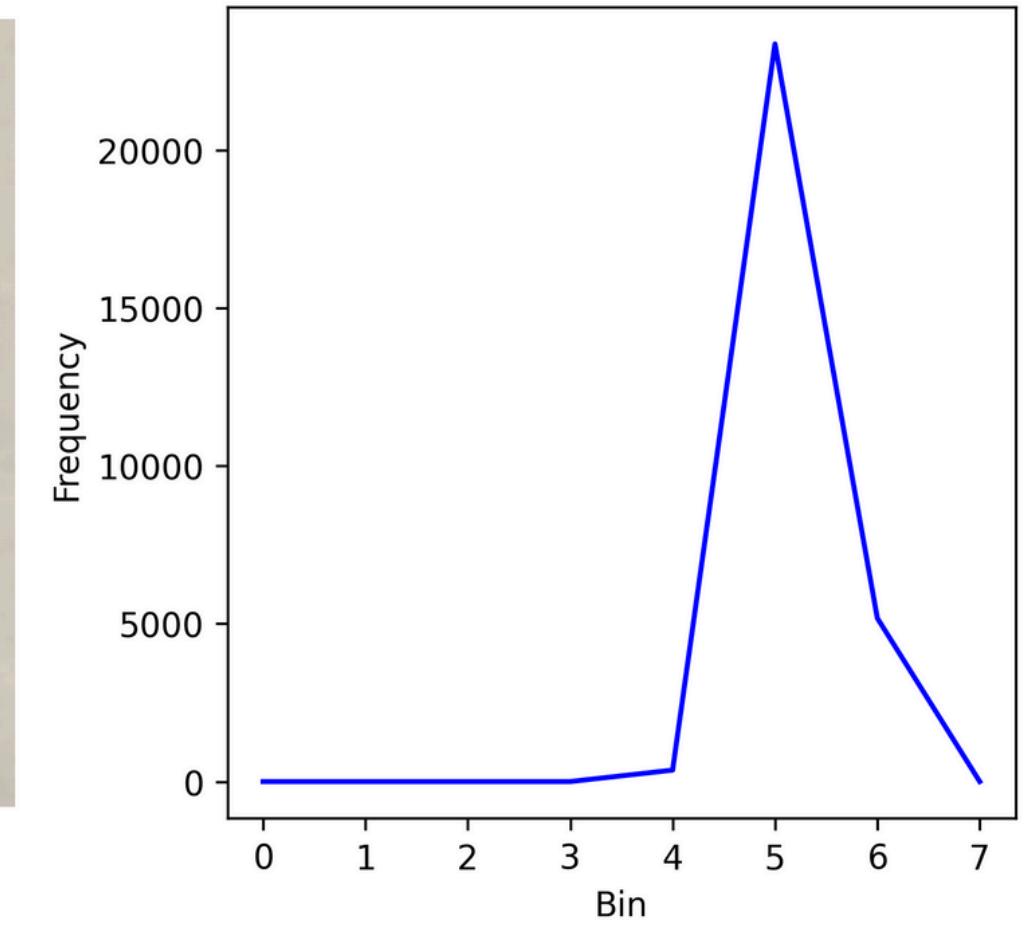
# Extraction of the histograms:



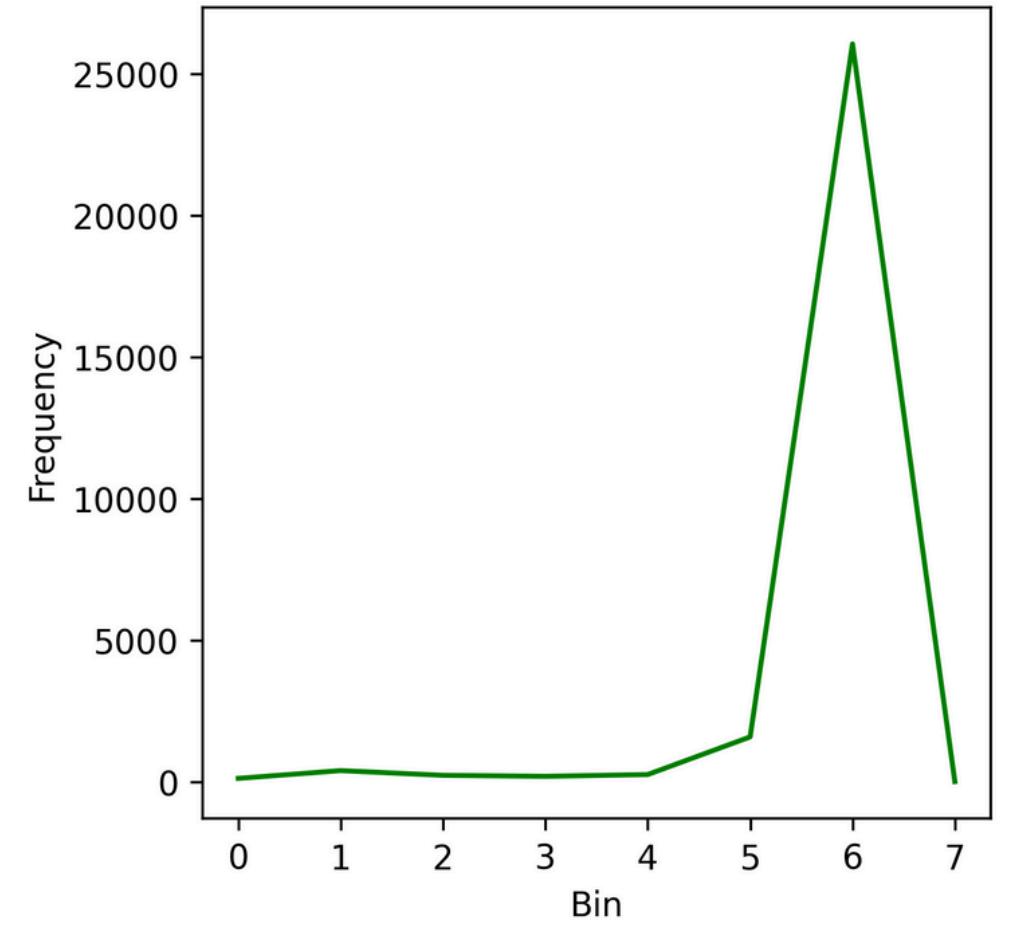
Original Image



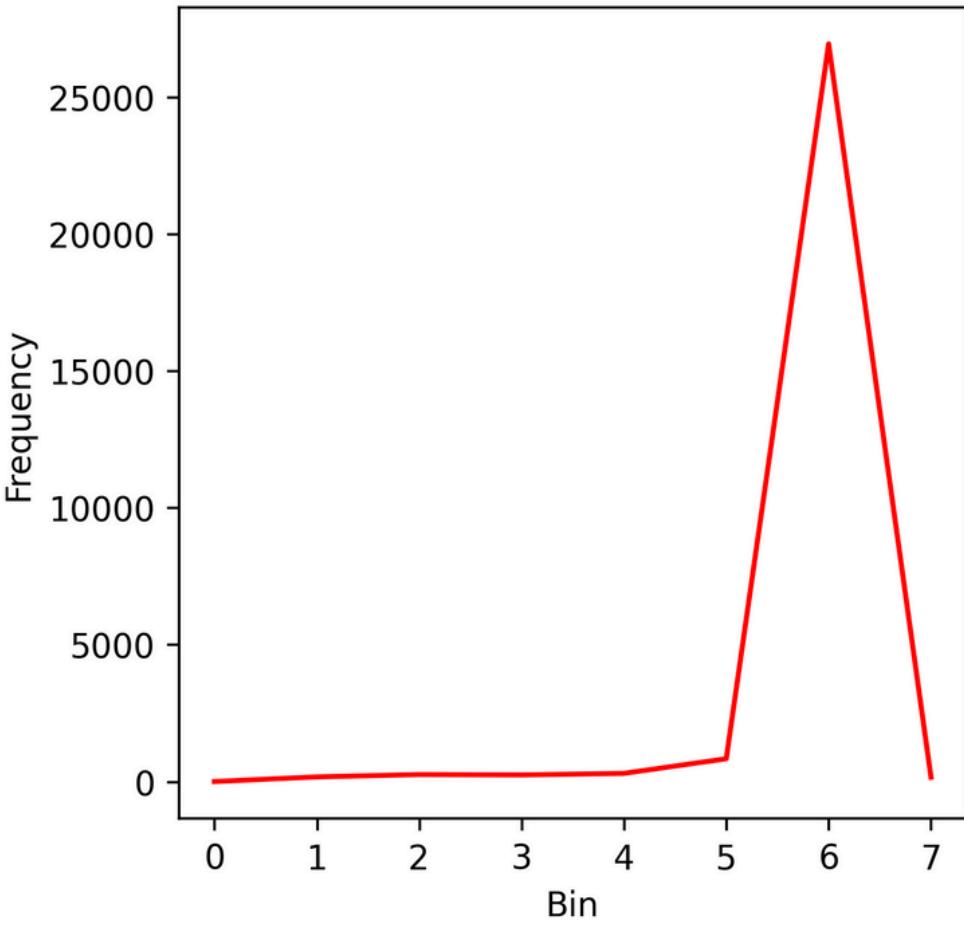
Blue Histogram



Green Histogram



Red Histogram



# Features Dataframe

Color Histogram - 24 features

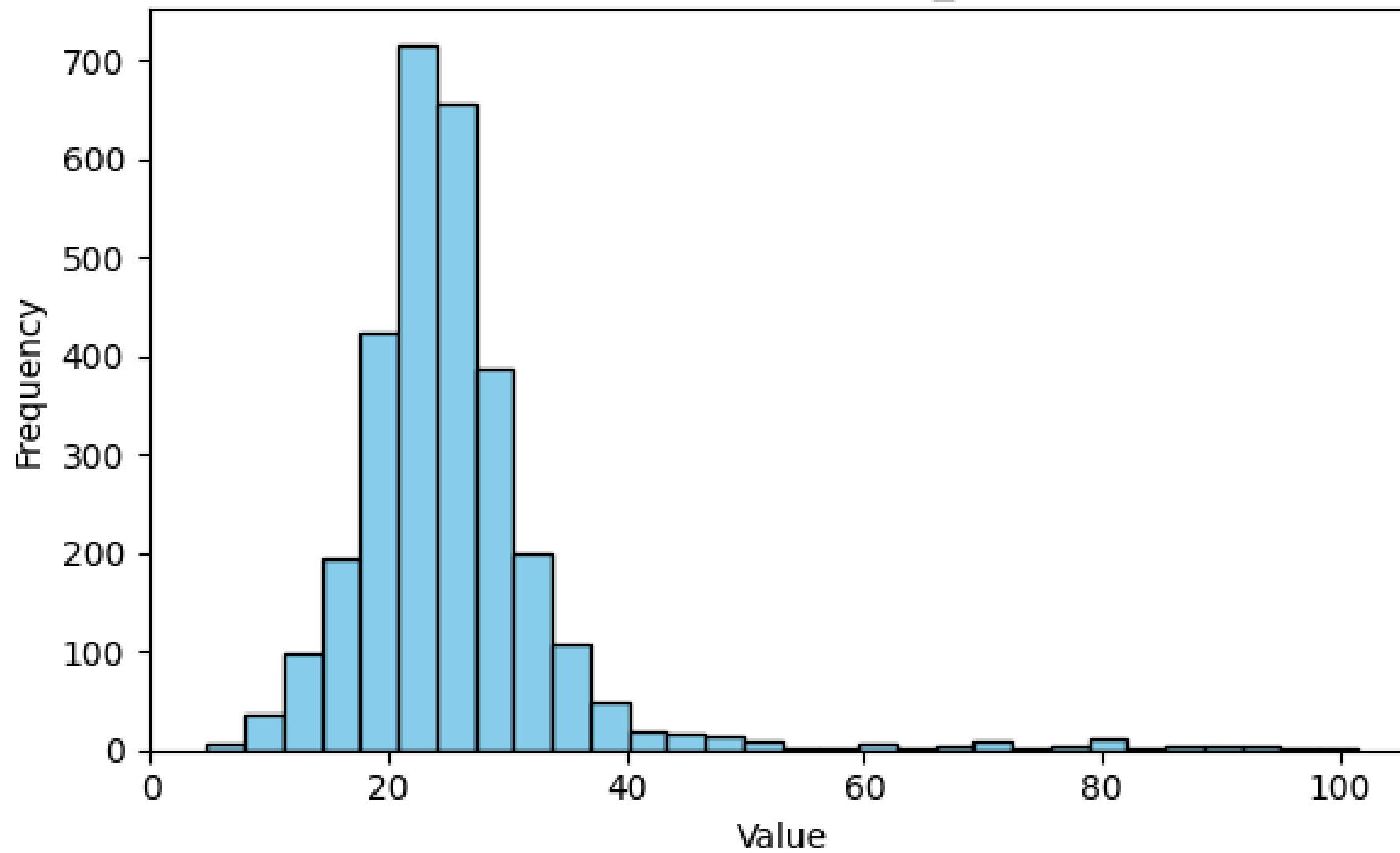
Blue_hist_0....7	Green_hist_0....7	Red_hist_0....7

Color Channels - 9 features

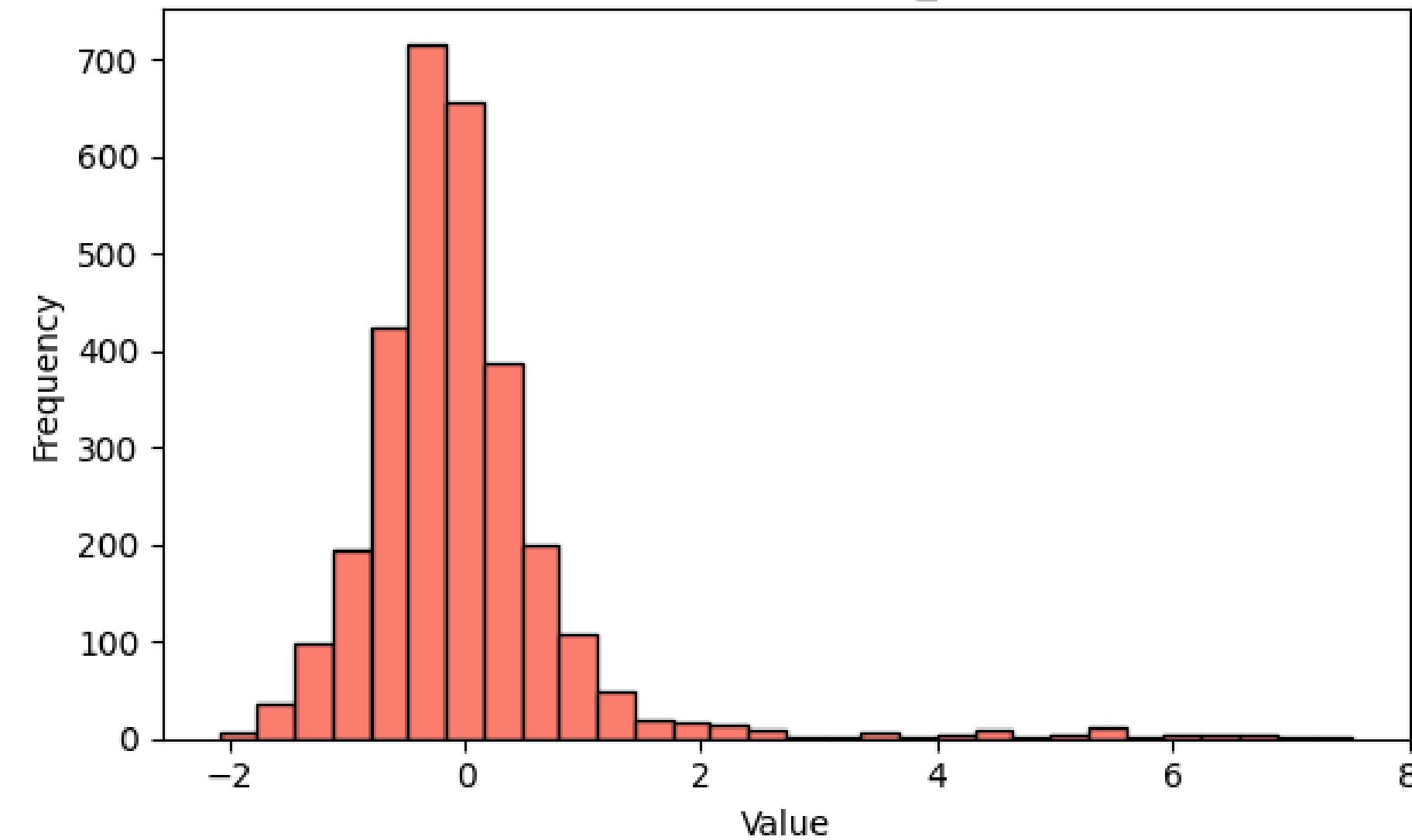
Mean(B,G,R)	Median(B,G,R)	Std. (B,G,R)

# STANDARDIZATION

Before Scaling: red\_std



After Scaling: red\_std



# Model training & Evaluation

# TRAIN-TEST SPLIT: PREPARING THE MODEL FOR EVALUATION

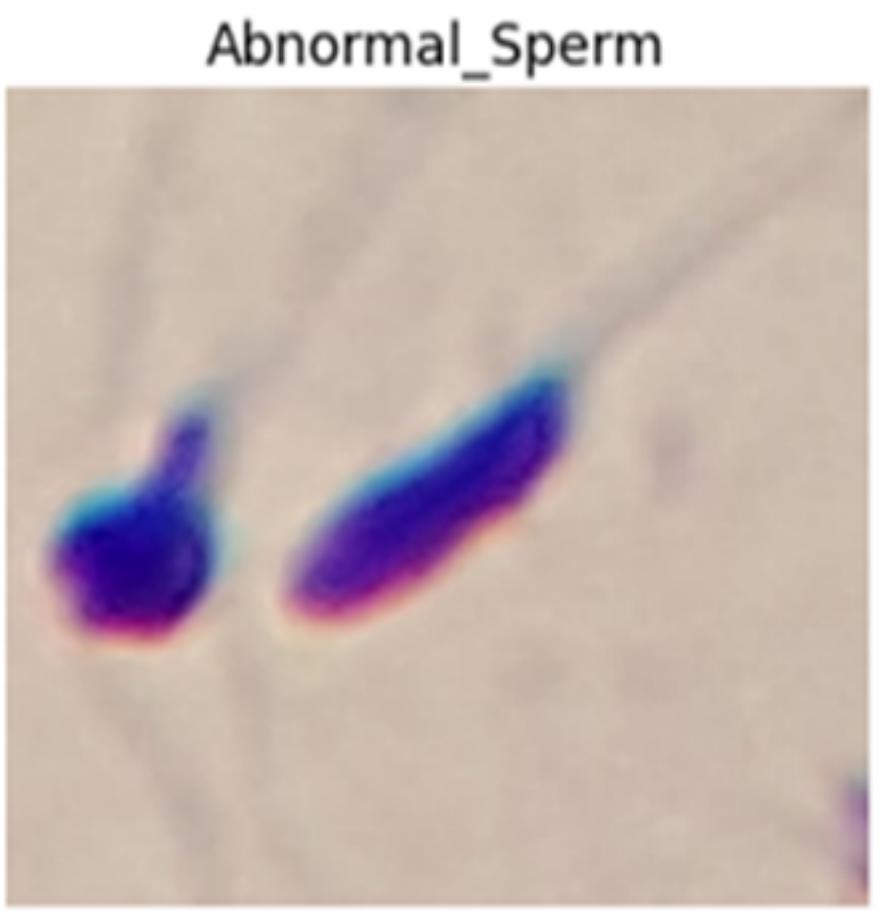


Sperm Morphology Image Dataset

**TRAIN SET**  
**80%**

**TEST SET**  
**20%**

# Targets



0

A large white number '0' is positioned on a dark blue background. A white arrow points from the number '0' to the 'Normal\_Sperm' image below it.

# KNN CLASSIFIER - THE 'PARAMETER FREE' CLASSIFIER



## **Default Values were applied in the implementation**

KNN was used with K=5, assigning each sample the majority class of its 5 nearest neighbors.

.....

Distance is measured using the Euclidean metric (L<sub>2</sub> Distance) .

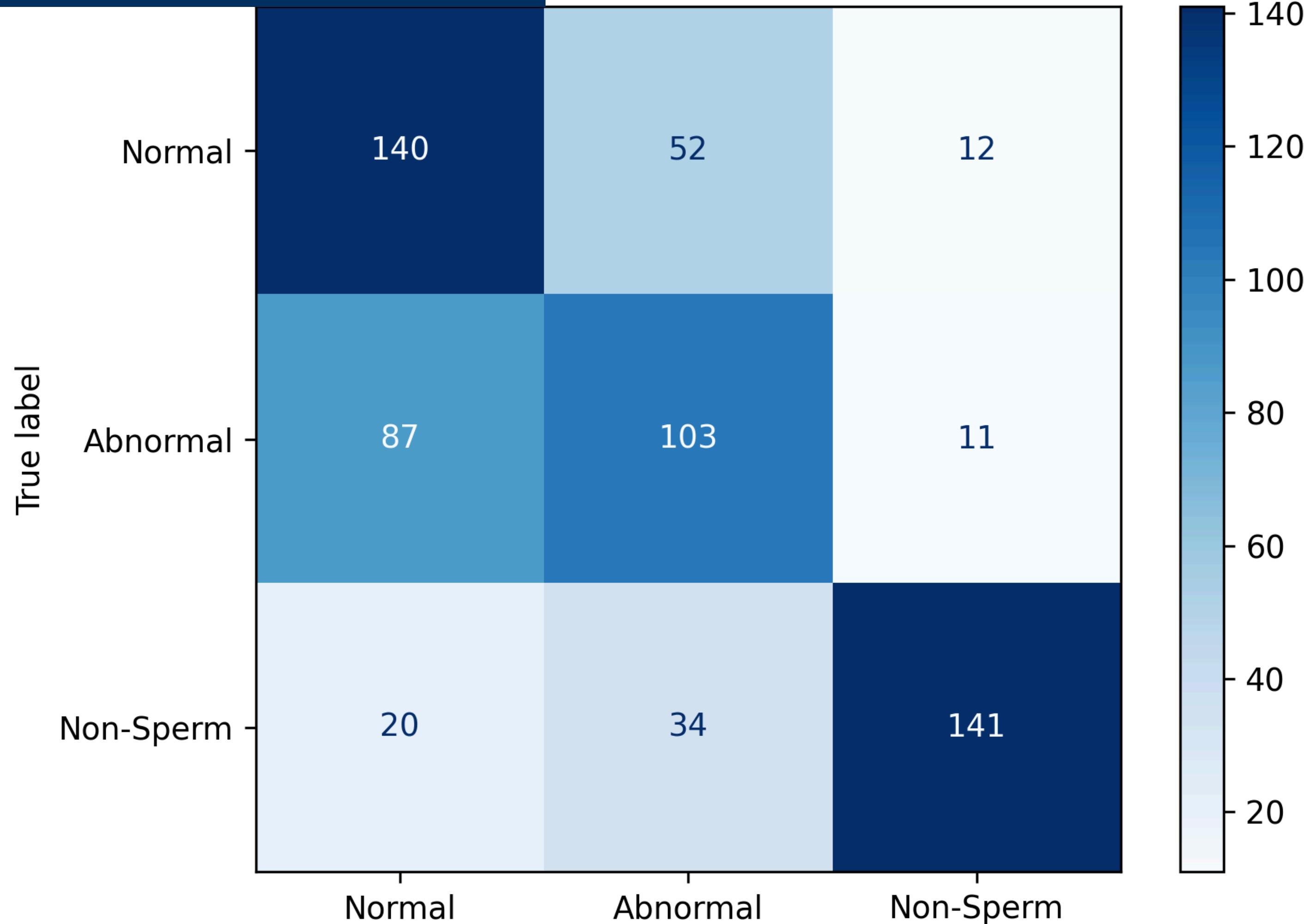
# EVALUATING THE KNN CLASSIFIER



**Question:**

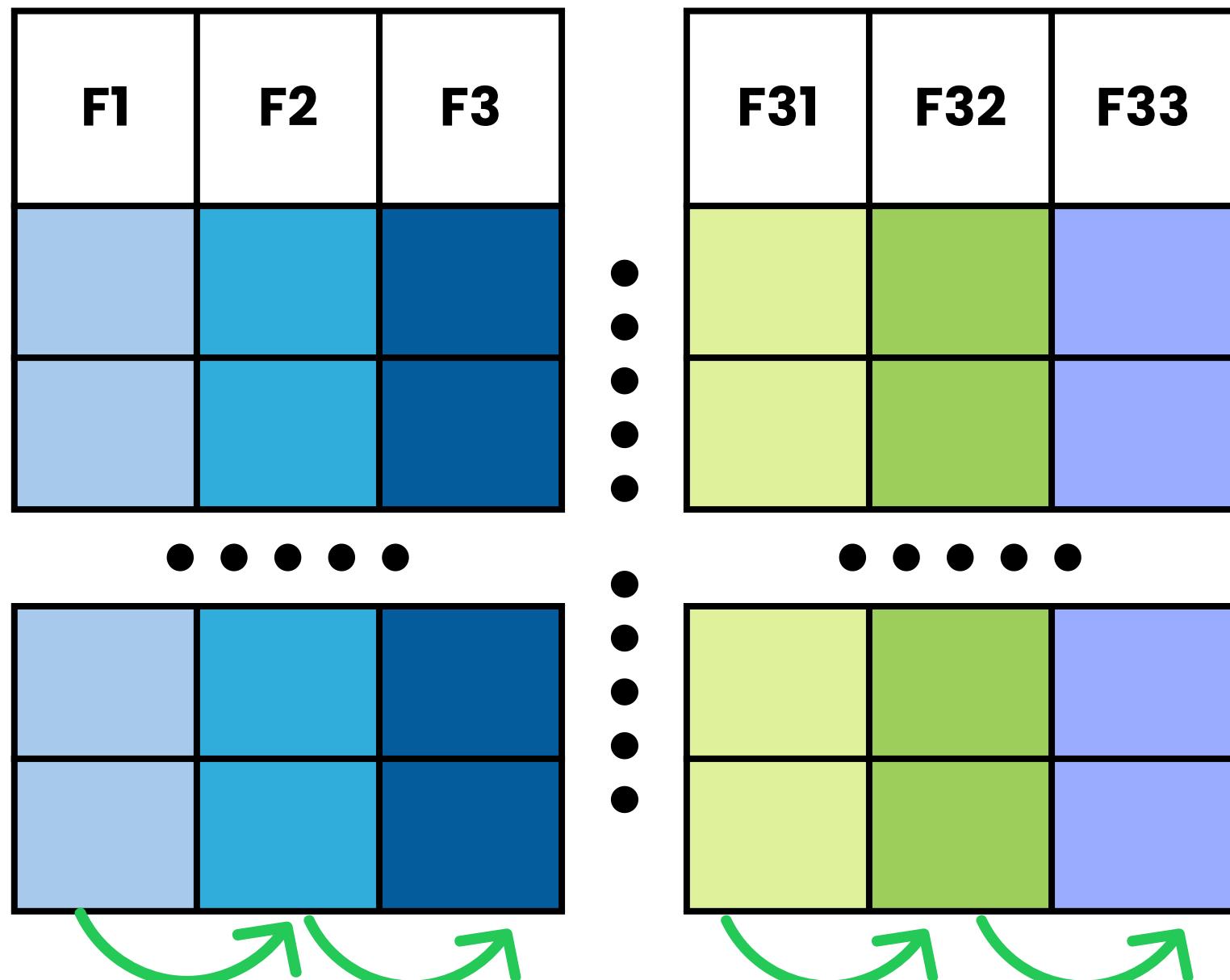
**What does the model get right — and where does it struggle?**

# CONFUSION MATRIX



# FORWARD FEATURE SELECTOR

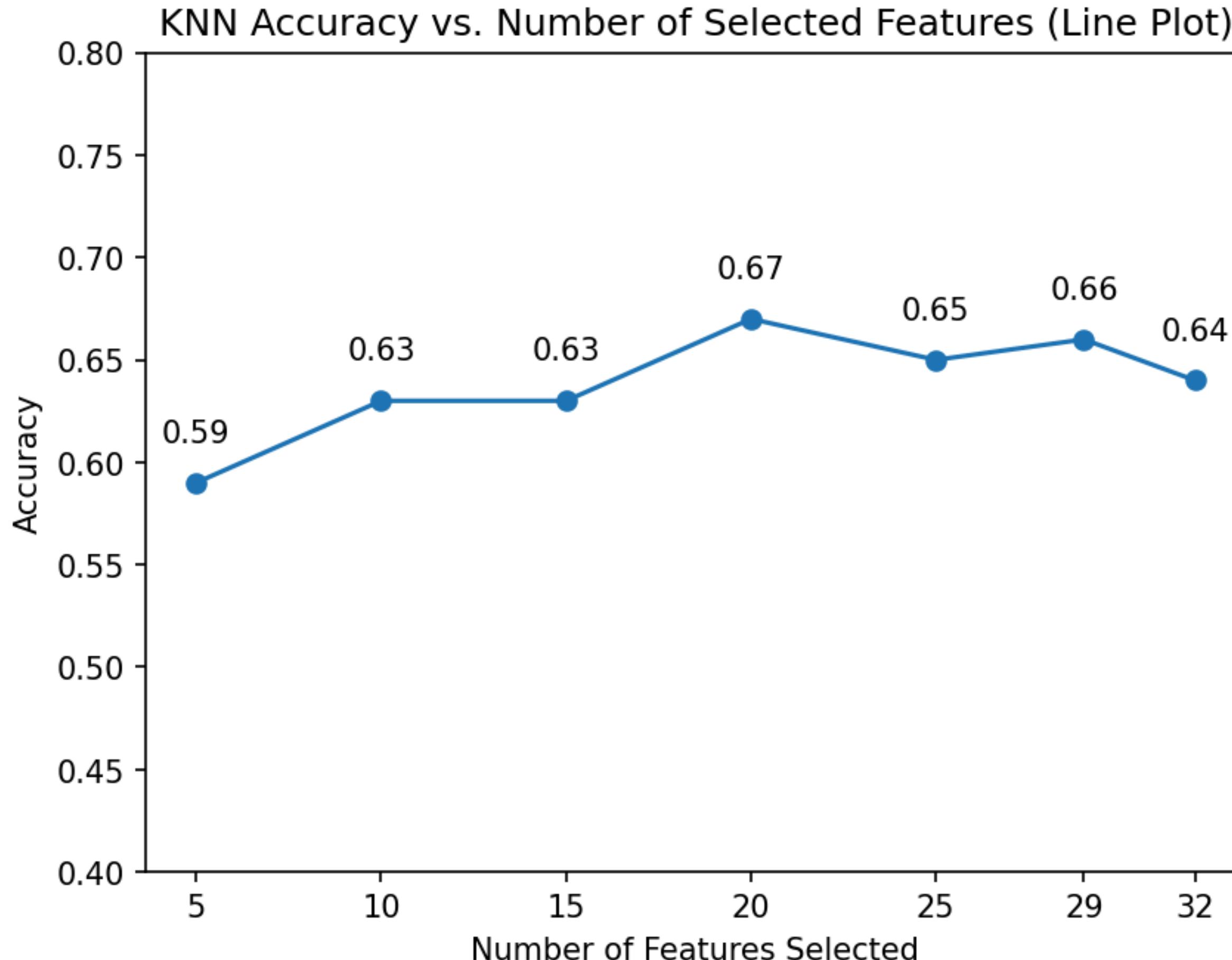
33 Features



Can we maximize our accuracy?

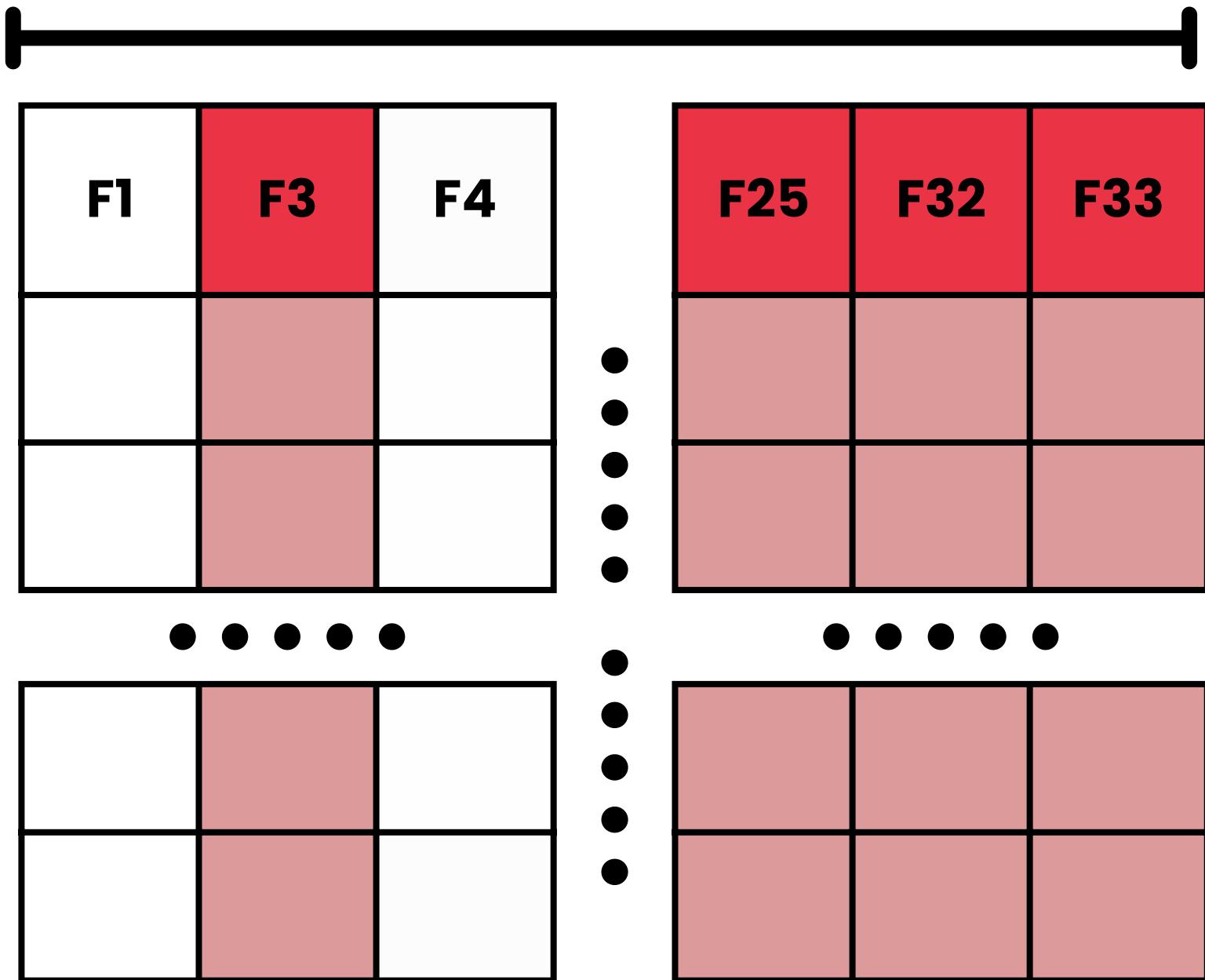
- Number of Features to select (for experimentation) :  
5, 10, 15, 20, 25, 29, 32
- Number of Features to select : 20
- Direction : forward
- Scoring Metric : Accuracy
- Cross-Validation : 5 folds

# EXPERIMENT: PERFORMANCE OF KNN ACCURACY ACROSS SELECTED FEATURES



# SFS EVALUATION : EVALUATION – RESULTS

## 20 Features

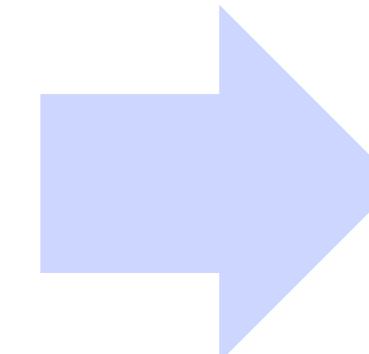


### Left-out features: 13

- blue\_median
- red\_median
- red\_hist\_5
- red\_hist\_6
- green\_hist\_5

### Accuracy:

0.64

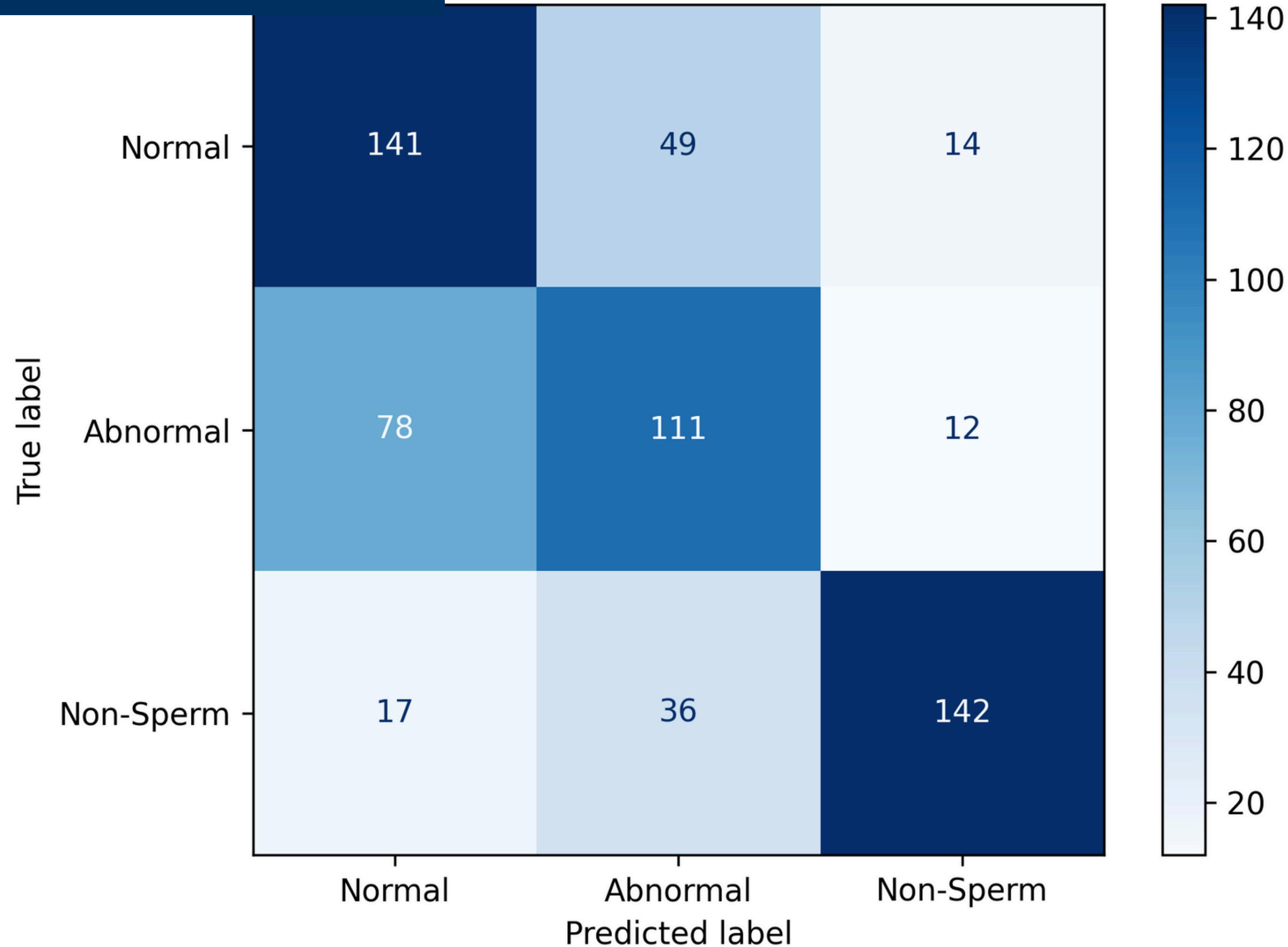


0.67

### Among Selected Features:

- blue\_hist\_7, red\_hist\_3, red\_hist\_2

# CONFUSION MATRIX



# SUPPORT VECTOR MACHINE – OPTIMIZED WITH GRIDSEARCHCV

**WHY**

**Good for average sized dataset,  
handles non-linear decision boundaries**

---

**GOAL**

---

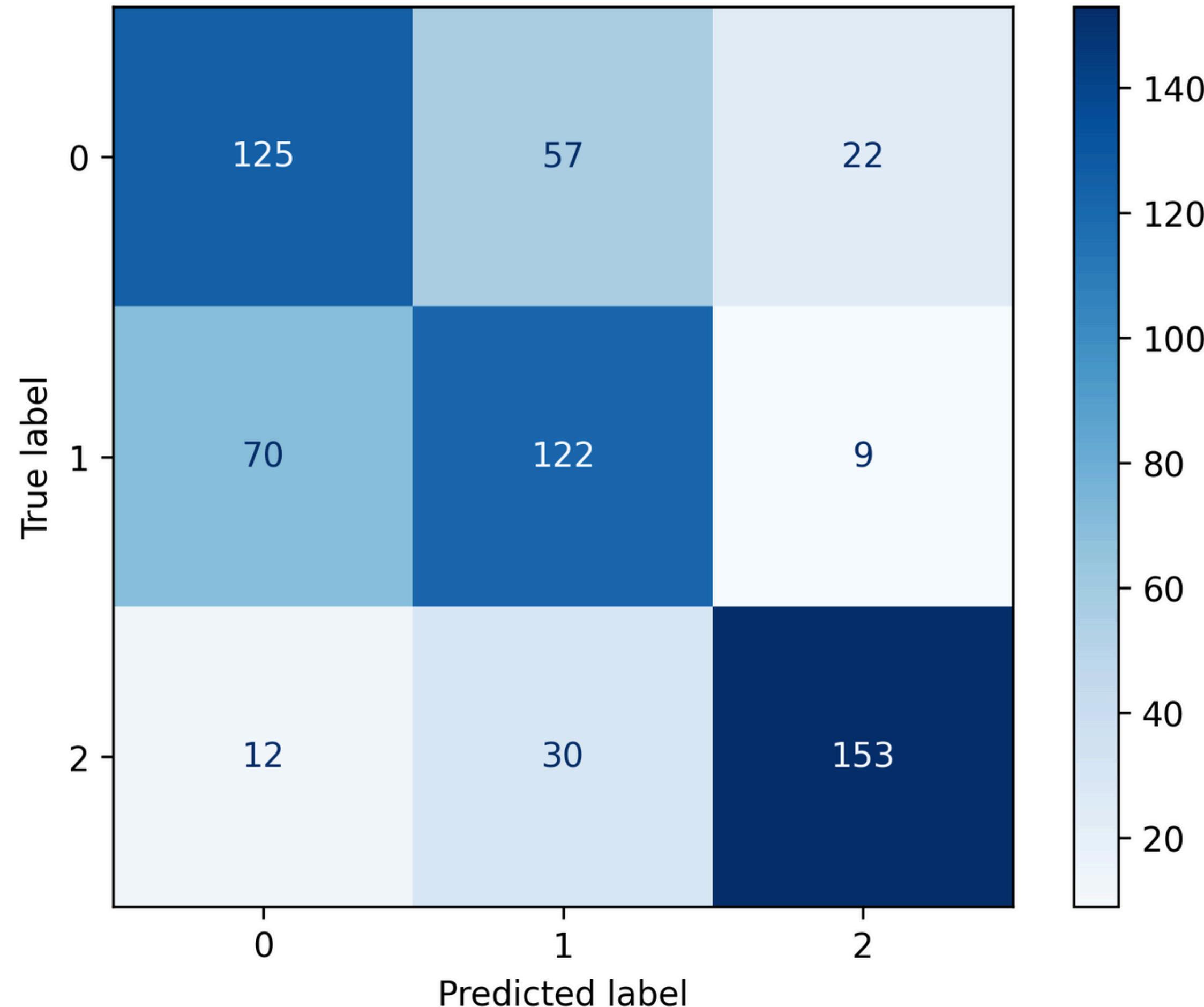
**To find the best parameters**

---

**Grid Search Combinations**

```
svm_params = {  
    'C': [0.01, 0.1, 1, 10, 100],  
    'kernel': ['rbf', 'linear' ]  
}
```

### SVM Confusion Matrix



# BEST PARAMETERS: FOR SUPPORT VECTOR MACHINE

C : 10

---

kernel: rbf

---

Accuracy: 0.67

# RANDOM FOREST CLASSIFIER – OPTIMIZED WITH GRIDSEARCHCV

**WHY**

Our dataset is not huge (3000 rows) , not too many iterations

**GOAL**

Tune the model by trying multiple parameter values and  
picking the top performer

## Grid Search Combinations

**n\_estimators = {50, 100, 150, 250, 300}**

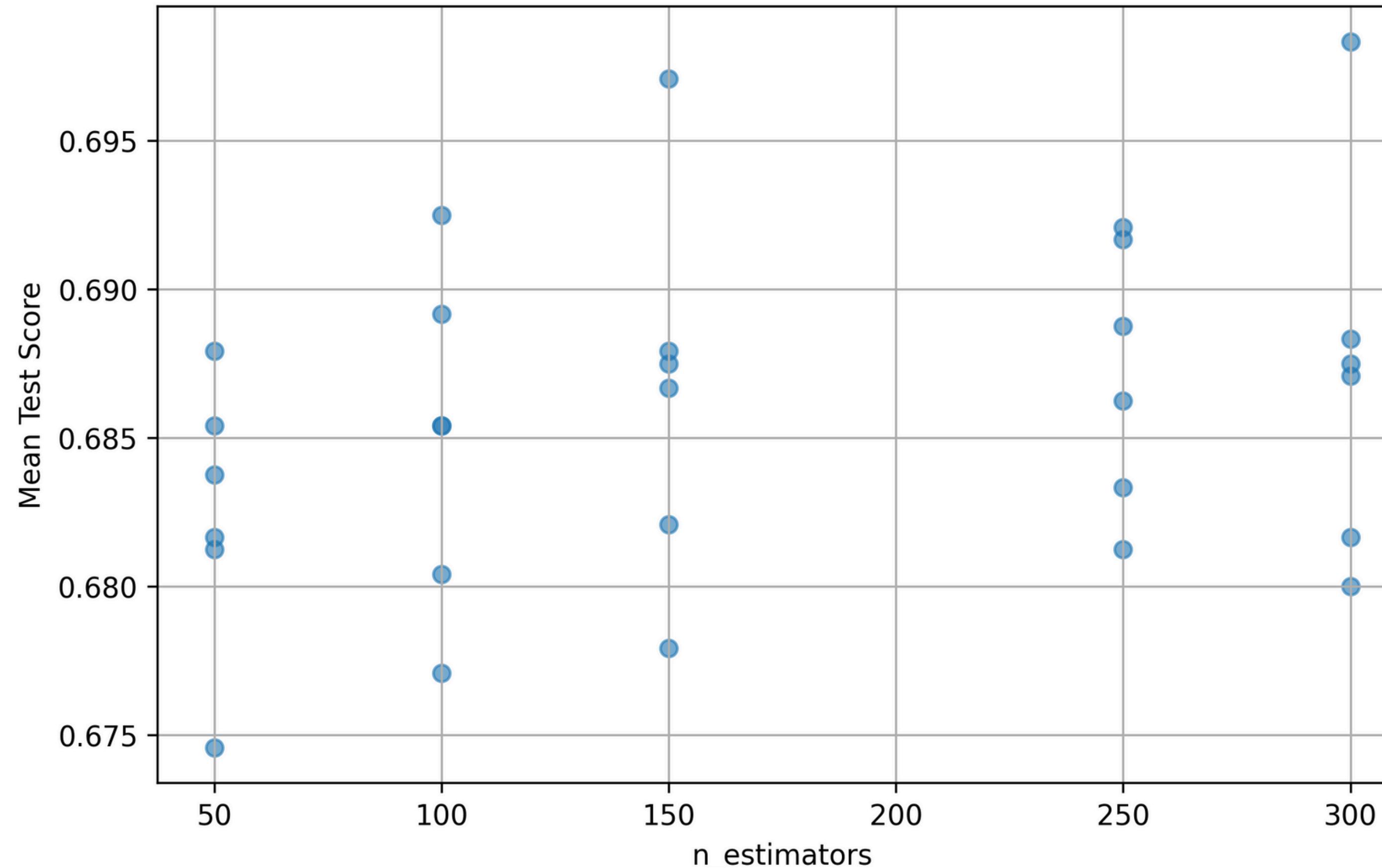
**max\_depth = {10, 20}**

**min\_samples\_split = {2,5}**

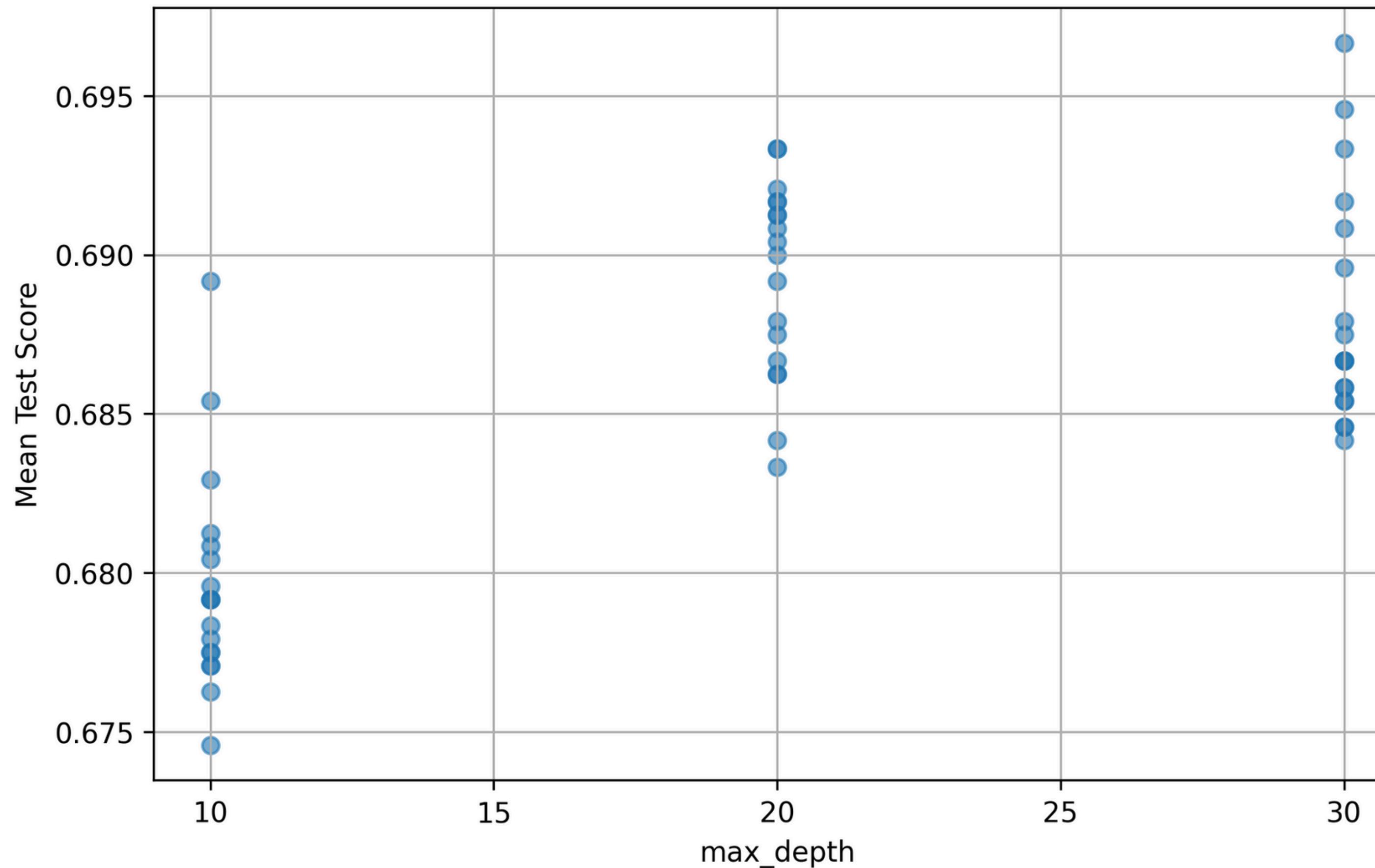
**min\_samples\_leaf = {1, 2}**

**max\_features = {None, 0.5, 0.8}**

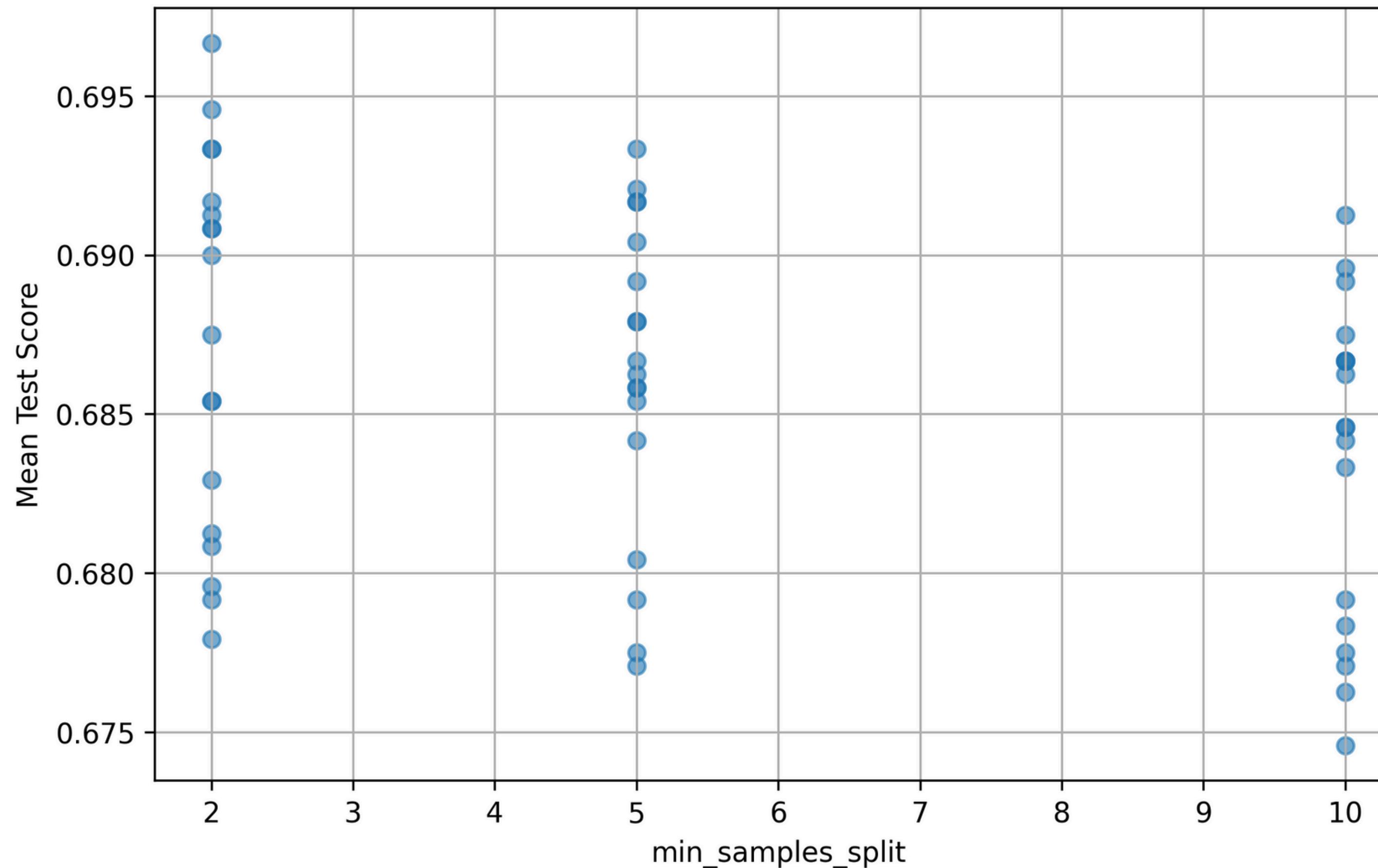
# MODEL PERFORMANCE VS N\_ESTIMATORS



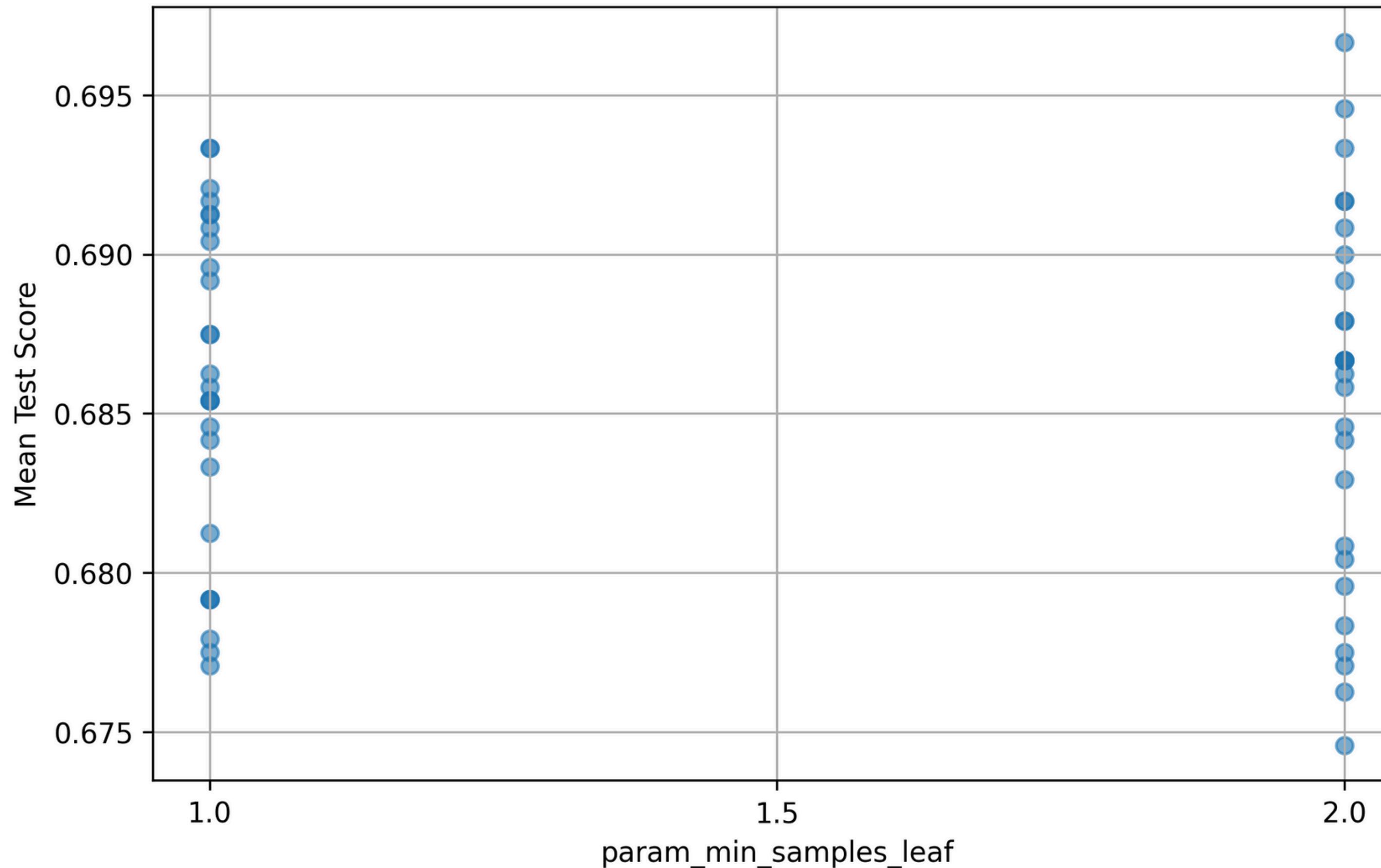
# MODEL PERFORMANCE VS MAX\_DEPTH



# MODEL PERFORMANCE VS MIN\_SAMPLES\_SPLIT



# MODEL PERFORMANCE VS MIN\_SAMPLES\_LEAF



## BEST PARAMETERS:

**max\_depth**

30

**min\_samples\_leaf**

2

**min\_samples\_split**

2

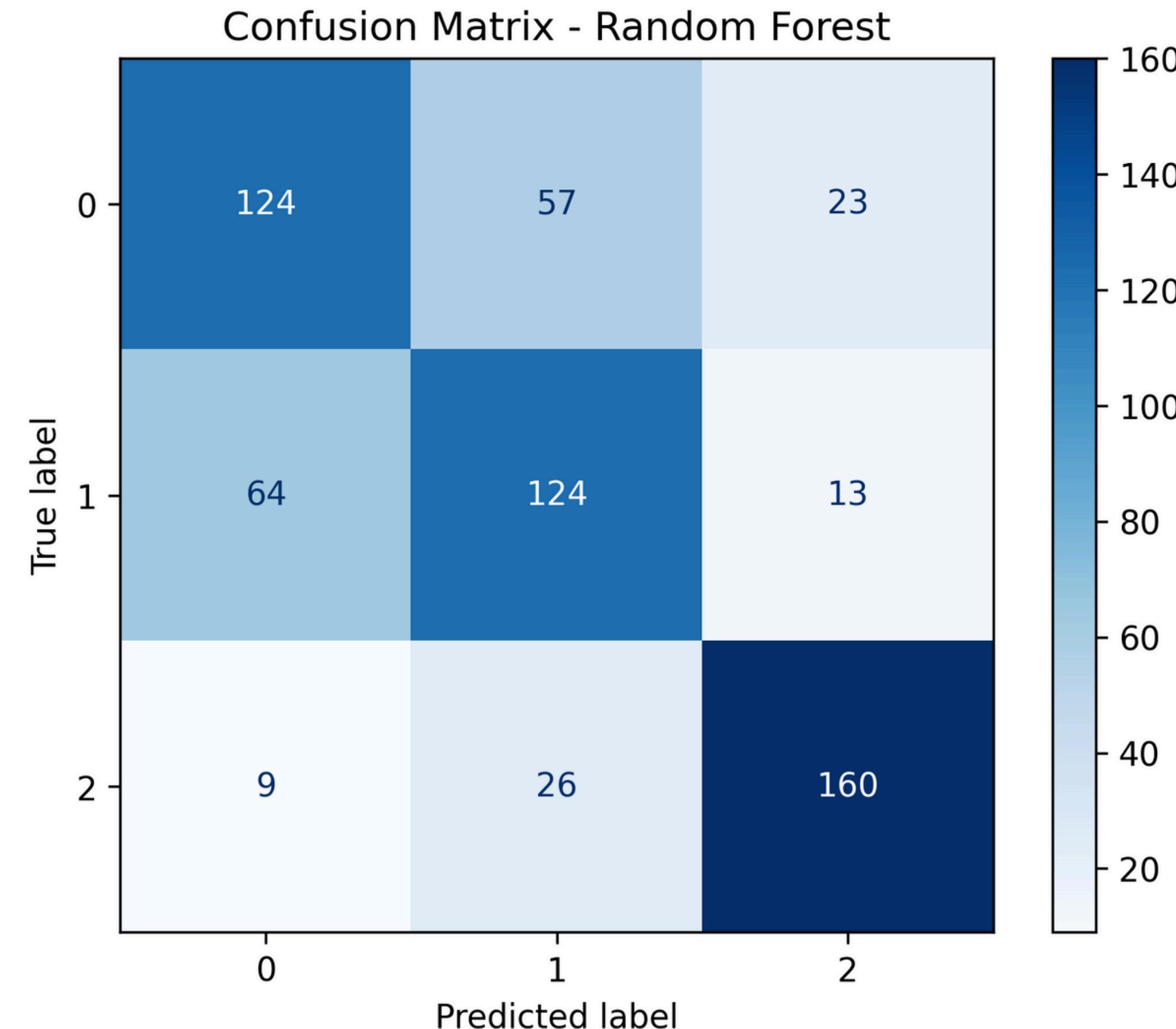
**n\_estimators**

200

**max\_features : 0.5**

**Accuracy: 0.68**

# CONFUSION MATRIX



## CONCLUSIONS:

- Models performance remains below 70% accuracy, even after feature selection and tuning.
- Misclassification between Normal Sperm and Abnormal Sperm.
- Feature Selection indeed increased the KNN model accuracy when selecting 20 features.
- Good for Binary classification of Sperms and Non-Sperms.
- Classification of Non-Sperm class is comparable to other advanced machine learning methods used on this datasets.