

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with diagonal stripes.

# Predictive Modelling

Group 2

# Outline:

Part I. Feature Extraction

Part II. Baseline Model Training

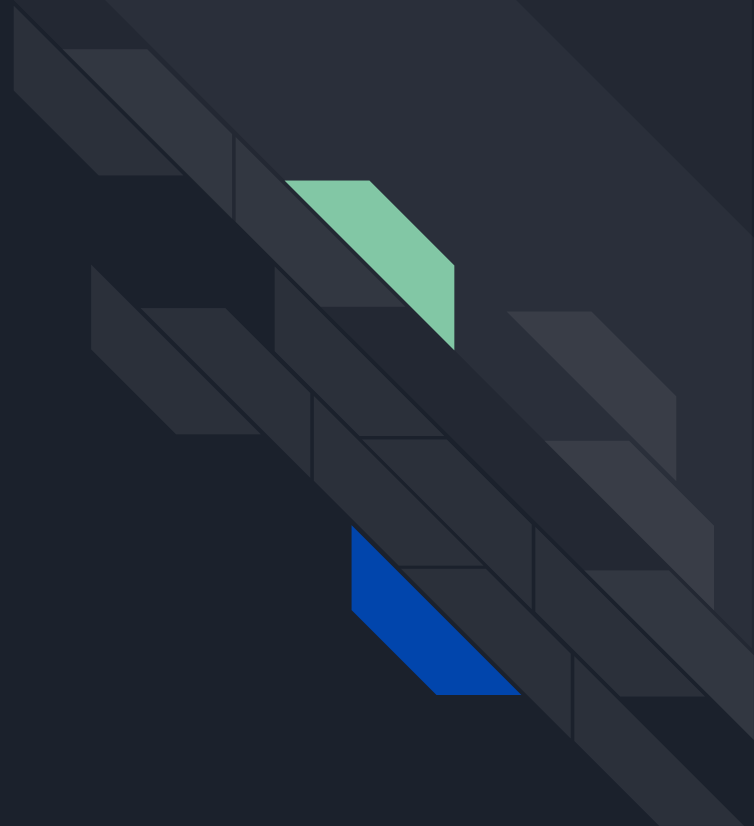
Part III. Advanced Model Selection

Part IV. Outcome Comparison



Part I.

Feature Extraction



## For featMat:

Green Color: Original Matrix

Red Color: Matrix after Parallel Translation

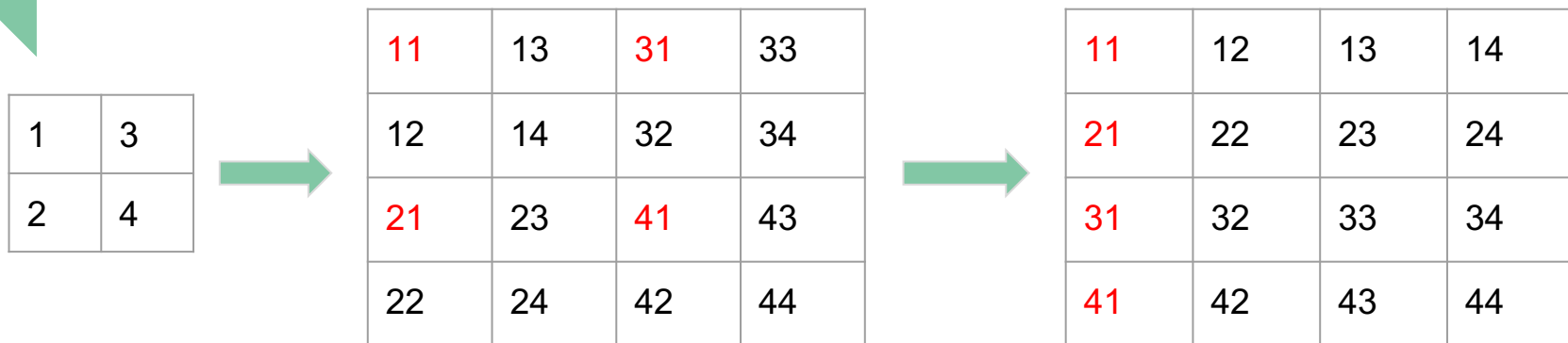
1	2	3
4	5	6
7	8	9

1	2	3	
4	5 1	6 2	3
7	8 4	9 5	6
	7	8	9



```
mat_1 <- abind(array(rep(0, row-1), c(row-1,1,3)), array(img[-row,-col,], c(row-1, col-1, 3)), along=2)
mat_1 <- abind(array(rep(0, col), c(1,col,3)), array(mat_1, c(row-1, col, 3)), along=1)
mat_1_channel <- mat_1[, ,color.i] - mat_base[, ,color.i]
#mat_1_channel <- mat_1[, ,color.i]
```

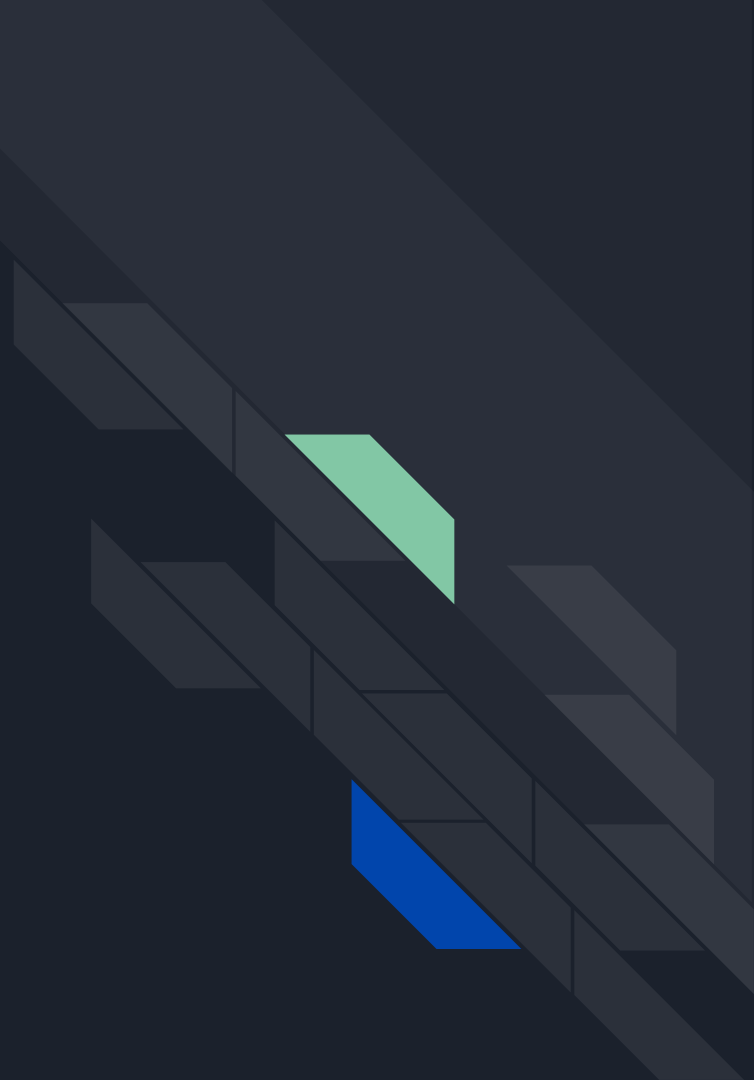
## For labMat:



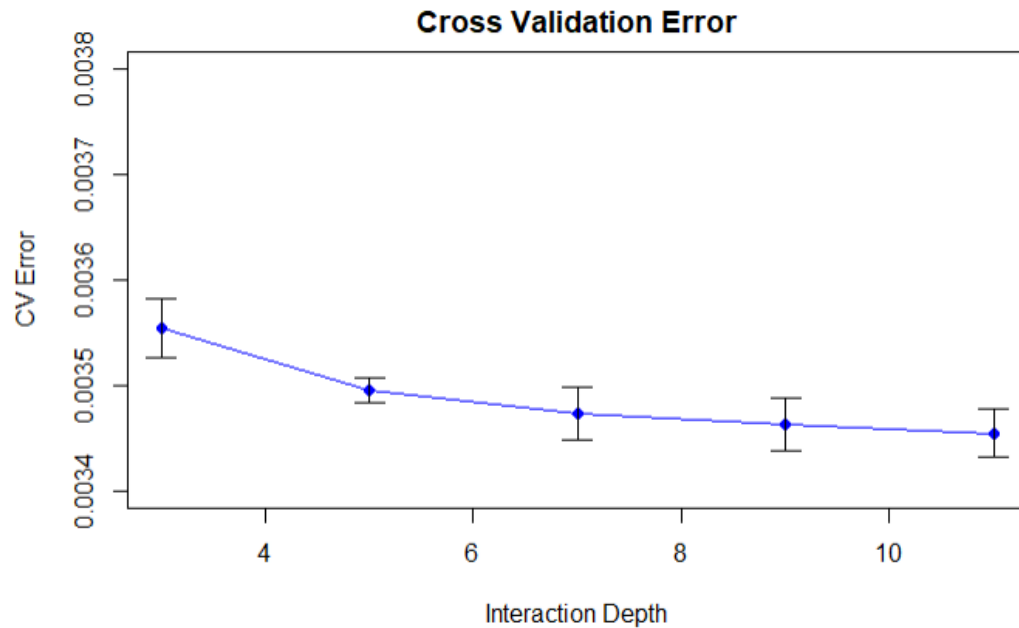
```
lab_mat[,1,] <- imgHR[seq(1,dim(imgHR)[1],2),seq(1,dim(imgHR)[2],2),]  
lab_mat[,2,] <- imgHR[seq(2,dim(imgHR)[1],2),seq(1,dim(imgHR)[2],2),]  
lab_mat[,3,] <- imgHR[seq(1,dim(imgHR)[1],2),seq(2,dim(imgHR)[2],2),]  
lab_mat[,4,] <- imgHR[seq(2,dim(imgHR)[1],2),seq(2,dim(imgHR)[2],2),]
```

Part II.

Baseline Model Training



For Baseline Model:



Potential Choices:  
5, 11

Optimal Depth:  
11

Part III.

# Advanced Model Selection



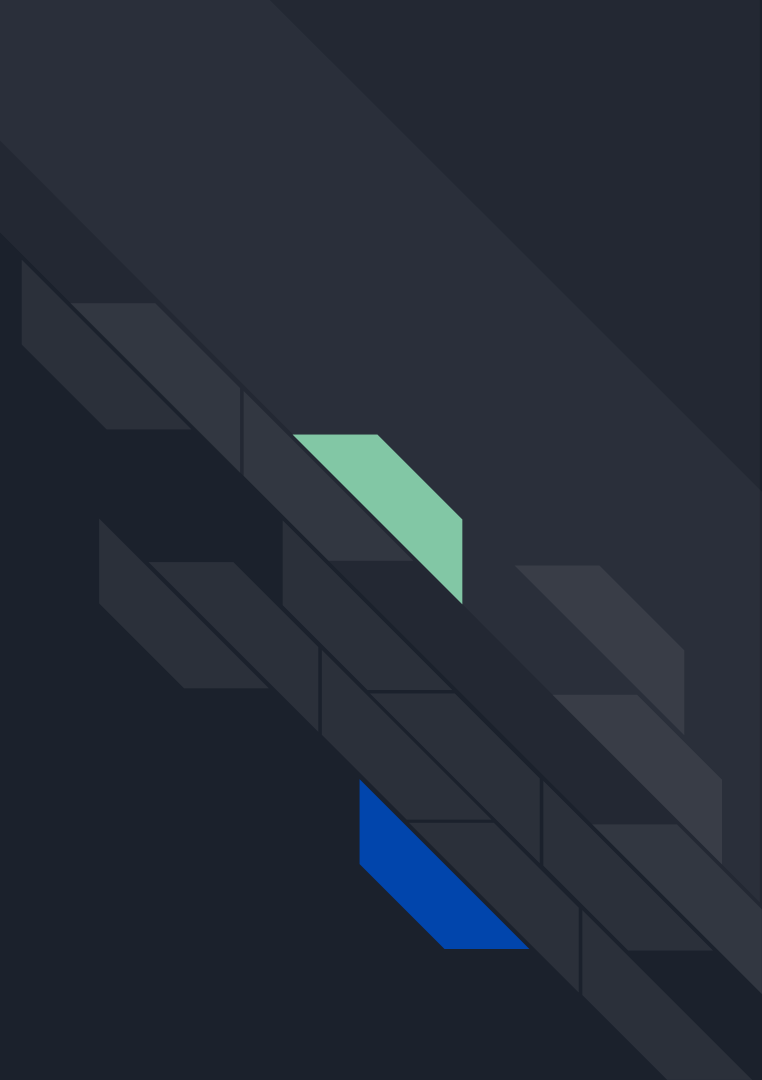


## For Advanced Model:

```
## xgboost model
xgboost <- NULL
if(run_xgb){
  if( !require("xgboost" )){
    install.packages("xgboost")
  }
  library("xgboost")
  #featMat <- feat_train[, , 1]
  #labMat <- label_train[, 1, 1]
  dtrain <- xgb.DMatrix(data=as.matrix(featMat),label=labMat)
  #xgboost_fit <- xgb.cv(data = dtrain, objective = "reg:linear",
  #                      metrics = "rmse",
  #                      eta = 0.5, max_depth = 8,
  #                      nthread= 2, nfold = 5, nrounds = 10)
  xgboost_fit <- xgboost(data = dtrain, objective = "reg:linear",
                        metrics = "rmse",
                        eta = 0.6, max_depth = 8,
                        nthread= 2, nfold = 5, nrounds = 10)
  modelList[[i]] <- list(fit=xgboost_fit)
}
```

Part IV.

Outcome Comparison





For 1500 training data::

For GBM Model:

```
> mean(MSE)
[1] 0.003413879
> sd(MSE)
[1] 0.002885994
```

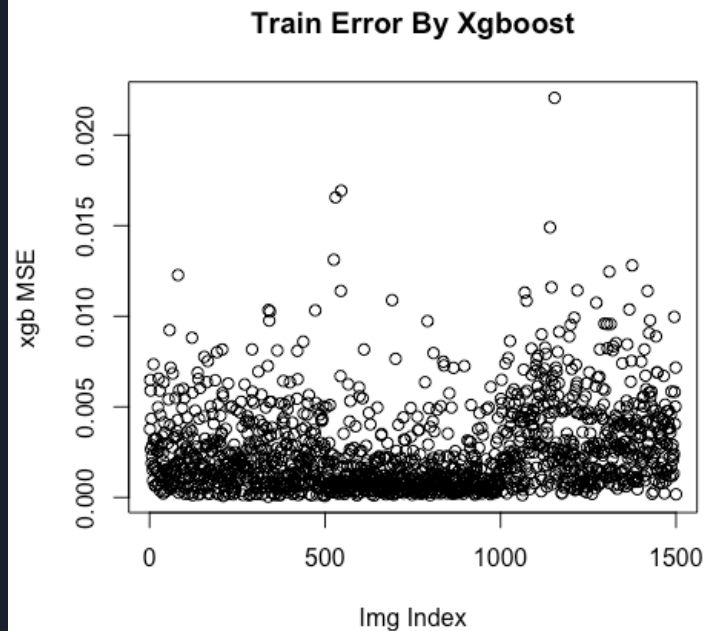
For XGBoost Model:

```
> mean(MSE)
[1] 0.002530288
> sd(MSE)
[1] 0.002383752
```

## For GBM Model



## For XGBoost Model





## Running Time Comparison:

	Time	Model Type
Tm_train	>6h	Training time of baseline
Tm_test	320s	Testing time of baseline
Tm_train_xgb	244s	Training time of xgb
Tm_test_xgb	36s	Testing time of xgb



For 50 Test Data:

For GBM Model:

```
> mean(MSE)
[1] 0.004563295
> sd(MSE)
[1] 0.002278968
```

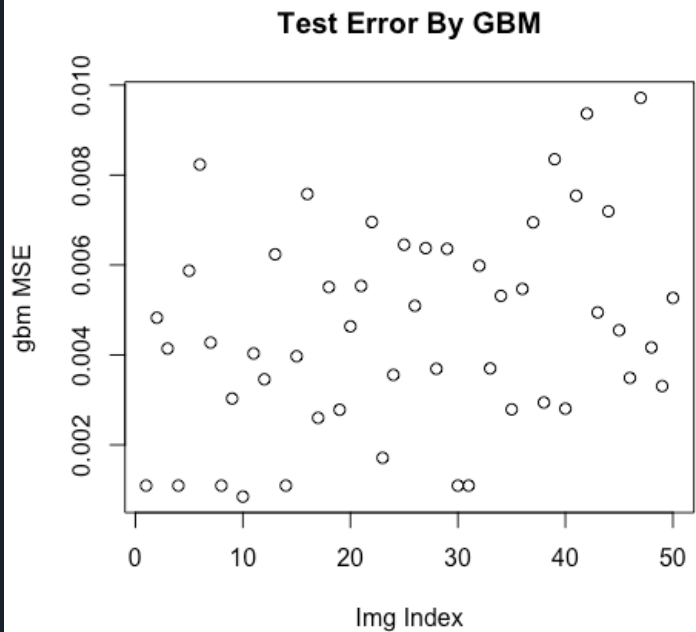
```
[1] 24.1222
> sd(PSNR)
[1] 2.785592
```

For XGBoost Model:

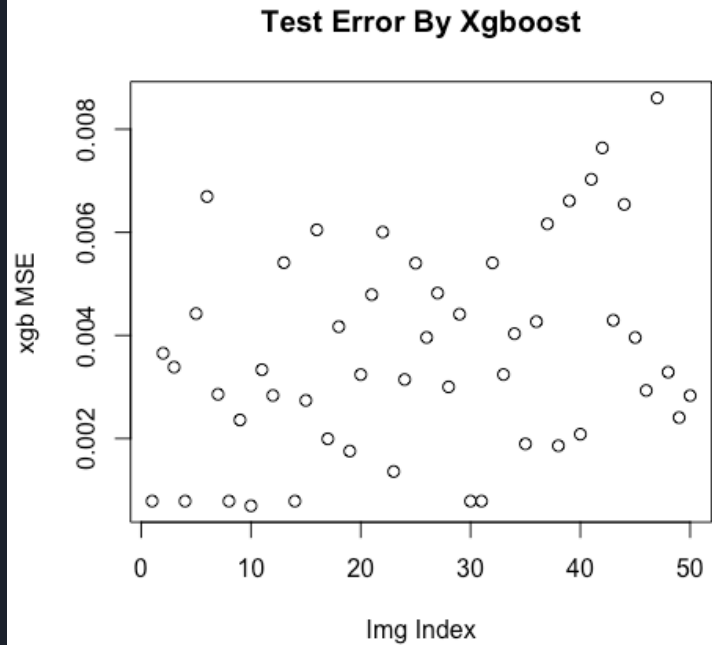
```
> mean(MSE)
[1] 0.003645735
> sd(MSE)
[1] 0.002002531
```

```
> mean(PSNR)
[1] 25.21666
> sd(PSNR)
[1] 2.991024
```

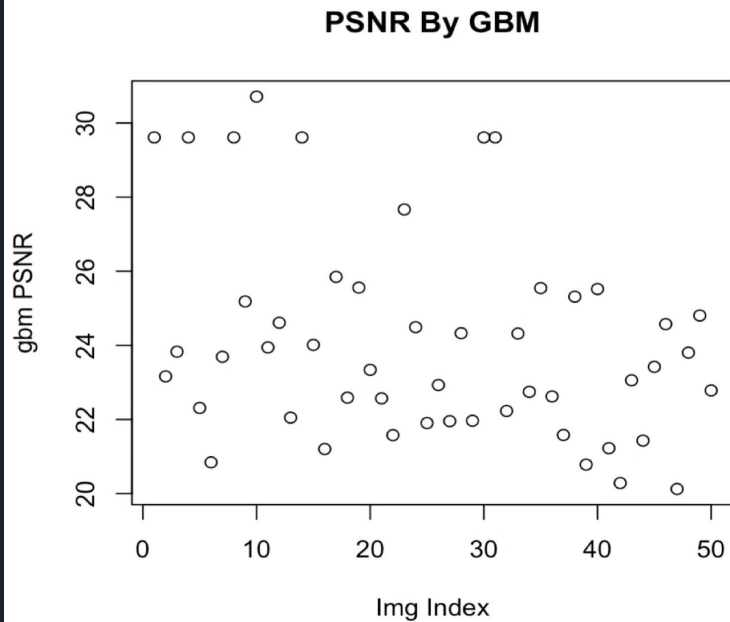
## For GBM Model



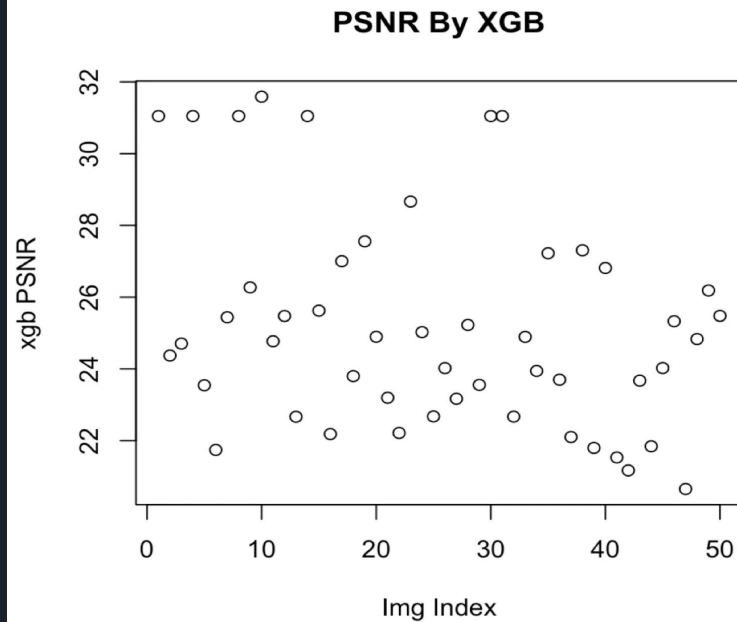
## For XGBoost Model



## For GBM Model



## For XGBoost Model

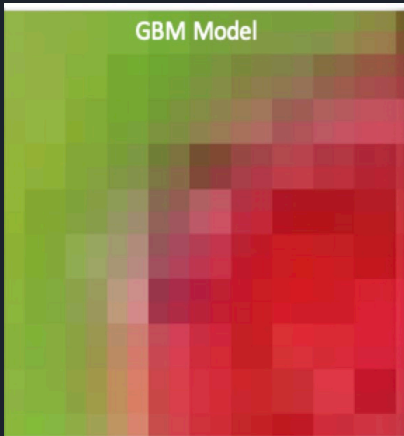




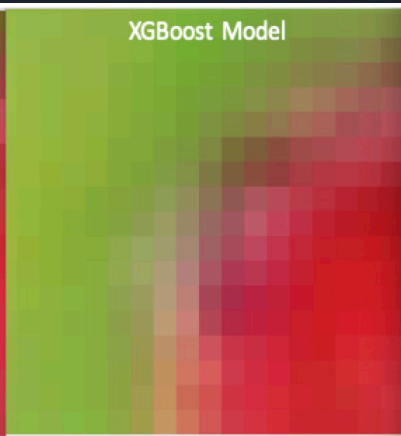
# Performance Comparison:



GBM Model



XGBoost Model



Ground Truth



Thank You!

