UNIVERSITY OF VIRGINIA

DATA SCIENCE INSTITUTE

Final Project Presentation

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Outline

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Executive Summary

Research Question:

- Can we create machine learning models to correctly predict if patients have a disease state?

Findings:

- Challenging to predict specific disease state, opted for normal vs disease state
- Most models explored were similar in value across key metrics
- Logistic Regression with a subset of features (both clinical and engineered) provided best results

Future Improvements:

- Better labelling of training data
- Some labelling of engineered features for comparison

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Data Summary

Data Set

Data Source: Physionet

Metadata

- Patient demographics, likelihoods for disease states, annotations, tools and personnel performing the ECG
- 34 columns in total
- Checked for duplicates
- Incomplete (values missing) - only “age” and “sex” deemed usable
- Outliers were not dropped as they could be strong indicators for disease state

ECG data

- 10 seconds of data for each ECG (at 100Hz, 500Hz)
- Across the 12 ECG leads
- No “missing” data, though some leads had no relevant information (flatline)

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Data Summary

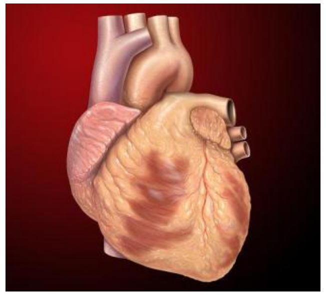
Response Variable

- Based on ECG data, patients were assigned a probability of having a disease state (multilabel) by a clinician
- Patients that were labelled with a value for probability disease probability (by a physician) were assigned a label for that disease

NORM	9528
MI	5486
STTC	5250
CD	4907
HYP	2655

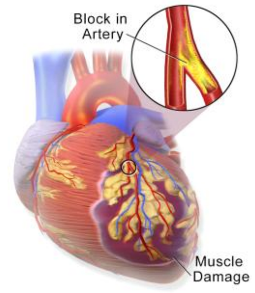
Table A: the relative counts for each respective label

NORM – Normal Heart



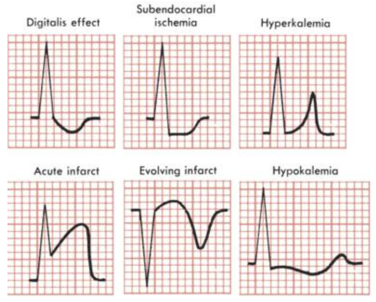
("Heart," 2020)

MI – Myocardial Infartion



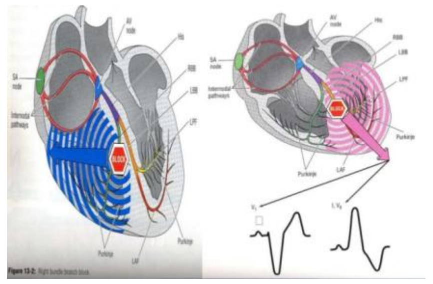
("Myocardial Infarction," 2020)

STTC – ST/T changes



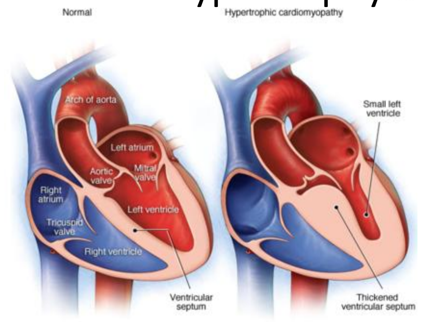
(*Drug Effects, Electrolyte Abnormalities, and Metabolic Factors - BASIC PRINCIPLES AND PATTERNS - Clinical Electrocardiography*, n.d.)

CD – Conduction Distrubance



(Jyotindra Singh, 12:01:51 UTC)

HYP - Hypertrophy



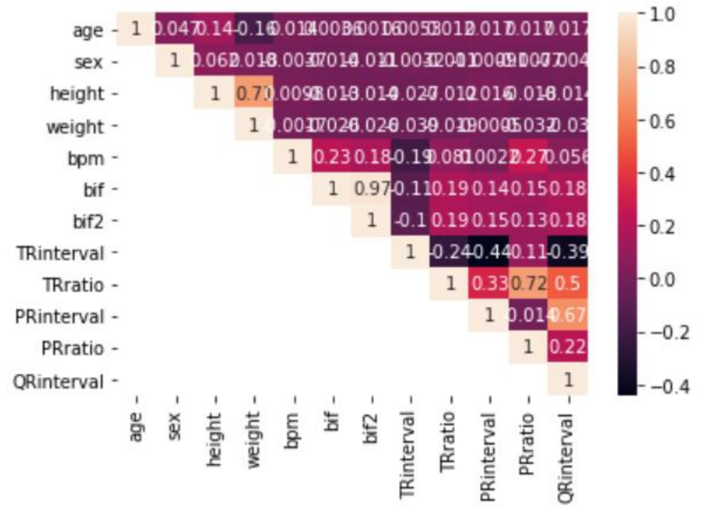
(*Hypertrophic Cardiomyopathy - Symptoms and Causes*, n.d.)

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Data Summary

Feature Subset Evaluation

Correlation matrix



Predictor	Outlier %
age	0.8
bpm	3.4
bif	16.2
bif2	15
TR interval	6.1
TR ratio	7
PR interval	9.7
PR ratio	6
QR interval	6.1

Feature Correlation

The engineered features were not highly correlated with each other, with the exception of TR ratio and PR ratio, as well as the PR interval and the QR interval. Though these features were somewhat correlated, each of them was considered important due their potential in identifying a disease state.

Feature Outliers

Outliers were not excluded, as strong outlier could actually denote a disease state, and capturing that state would be important for the model

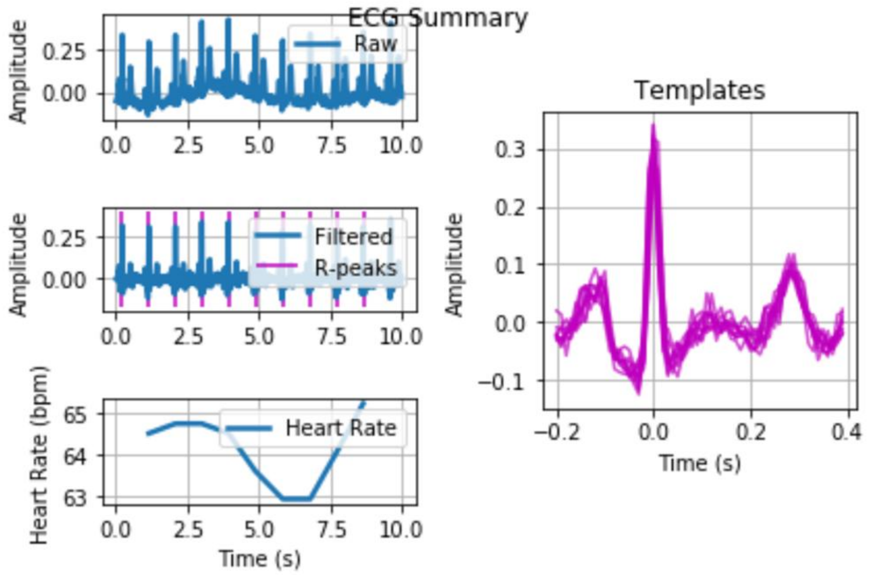
UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Variable Preprocessing

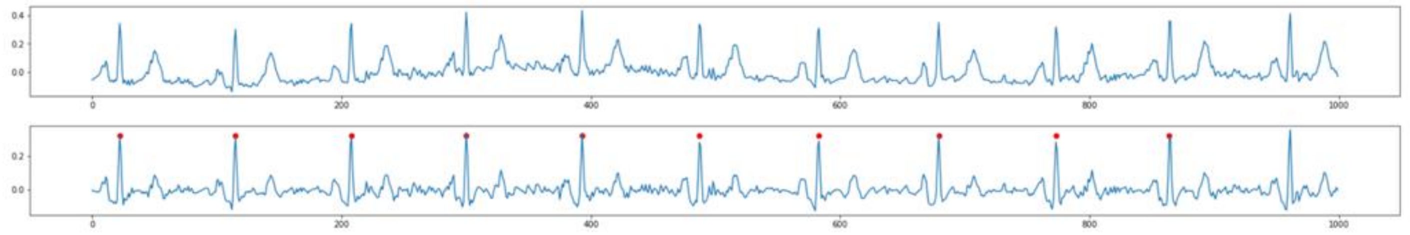
ECG Filtering, R peak calculation

- Needed to filter in order to calculate QRS complex, P wave, and T wave locations
- After many original attempts, the Biosppy package used
- Multiple edge cases evaluated for filtering performance
- Heart beat irregularity factor: range and standard deviation of heart beat intervals

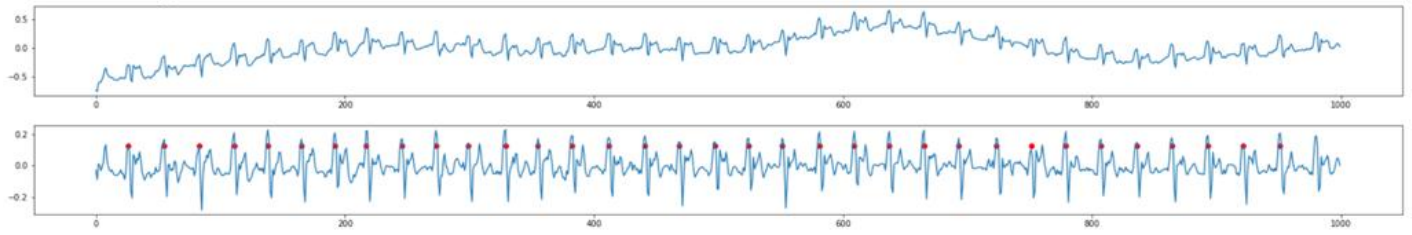
ECG Summary



Normal ECG



ECG with high heart rate and noise from power input

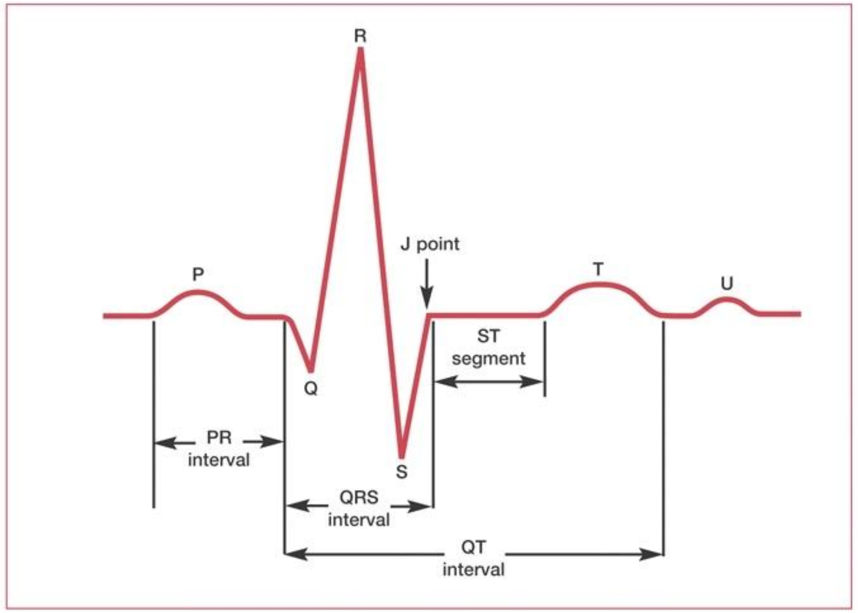


UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

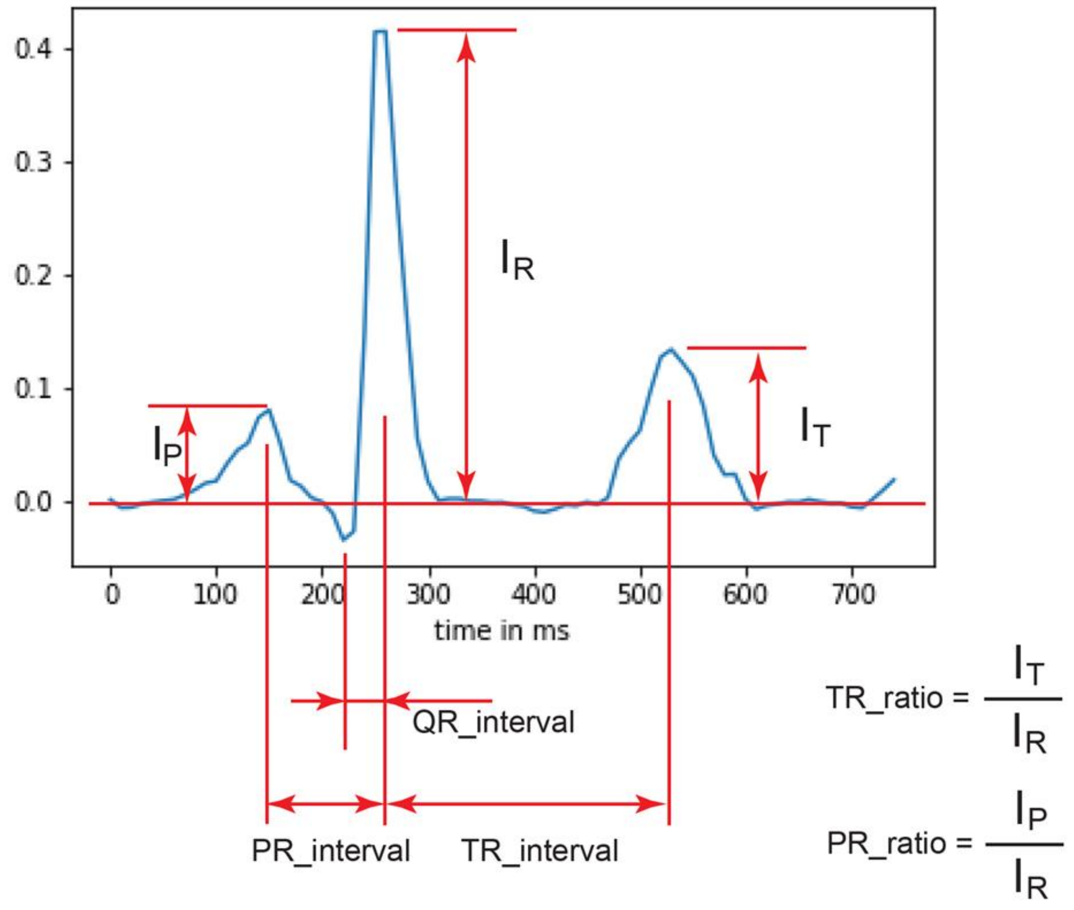
Variable Transformation

Feature Engineering

Plot 1: Annotated PQRST Complex



Plot 2: Features Extracted from Raw ECG Data



$$\text{TR\_ratio} = \frac{I_T}{I_R}$$

$$\text{PR\_ratio} = \frac{I_P}{I_R}$$

Engineered features: Beats per Minute, bif, bif2, TRinterval, TRratio, PRinterval, PRratio, QRinterval

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Pipeline and Cross Validation

Pipeline

- VectorAssembler
- StandardScaler
- Selected Models, such as Logistic Regression, Random Forest, Gradient Boosted Tree, and Support Vector Machine

Cross Validation

- Pipeline as estimator
- Two hyperparameter grid
  --varies from 12 - 36
- BinaryClassificationEvaluator
  -- AUROC
- 10 fold CV

Code example

```
# Configure an ML pipeline,three stages: assembler, standardScaler, and svm.
assembler = VectorAssembler(
                    inputCols=["age","sex", "bpm", "bif", "bif2"],
                    outputCol="features")
standardScaler = StandardScaler(inputCol="features", outputCol="features_scaled",
                    withStd=True, withMean=False)
svm = LinearSVC(featuresCol="features_scaled")
pipeline = Pipeline(stages=[assembler, standardScaler, svm])


# Set up the parameter grid
paramGrid = ParamGridBuilder() \
    .addGrid(svm.maxIter, [20,40,60]) \
    .addGrid(svm.regParam, [0.25,0.1,0.01,0.001]) \
    .build()

# Treat the pipeline as an Estimator, wrapping it in a CrossValidator instance.
crossval = CrossValidator(estimator=pipeline,
                  estimatorParamMaps=paramGrid,
                  evaluator=BinaryClassificationEvaluator(),
                  numFolds=10,
                  parallelism=10)

# Run cross-validation, and choose the best set of parameters.
cvModel = crossval.fit(train)
```

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

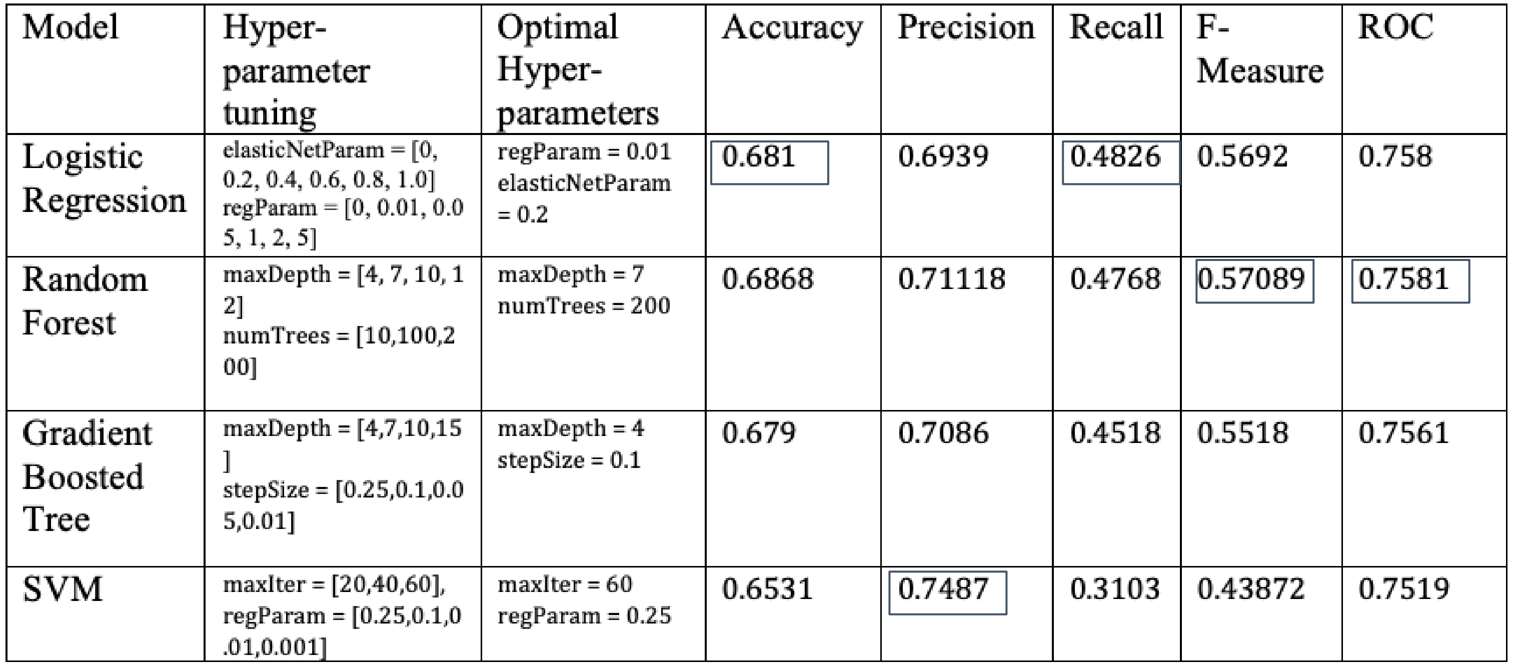
Iteration 1: Initial Model Exploration and Results

- Logistic regression, decision trees, random forest, gradient-boosted tree, linear SVM
- Outcome: NORM, MI, STTC, CD, HYP
- Feature selection: sex, age, all feature engineered predictors
- Features were scaled
- Stratfolds 9 and 10 were reserved for testing based on physician's knowledge in data documentation
- Accuracy ranged between 59% to 87%. Precision could not be calculated at times due to lack of true positive and false positive values. Decision trees and logistic regression especially performed poorly for most outcome variables with respect to recall and f-measure.

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Iteration 2: Model Tuning and Results

- Logistic regression, decision trees, random forest, gradient-boosted tree, linear SVM
- Outcome: NORM versus not NORM
- Feature selection: sex, age, all feature engineered predictors
- Features were scaled
- Stratfolds 9 and 10 were reserved for testing based on physician’s knowledge in data documentation

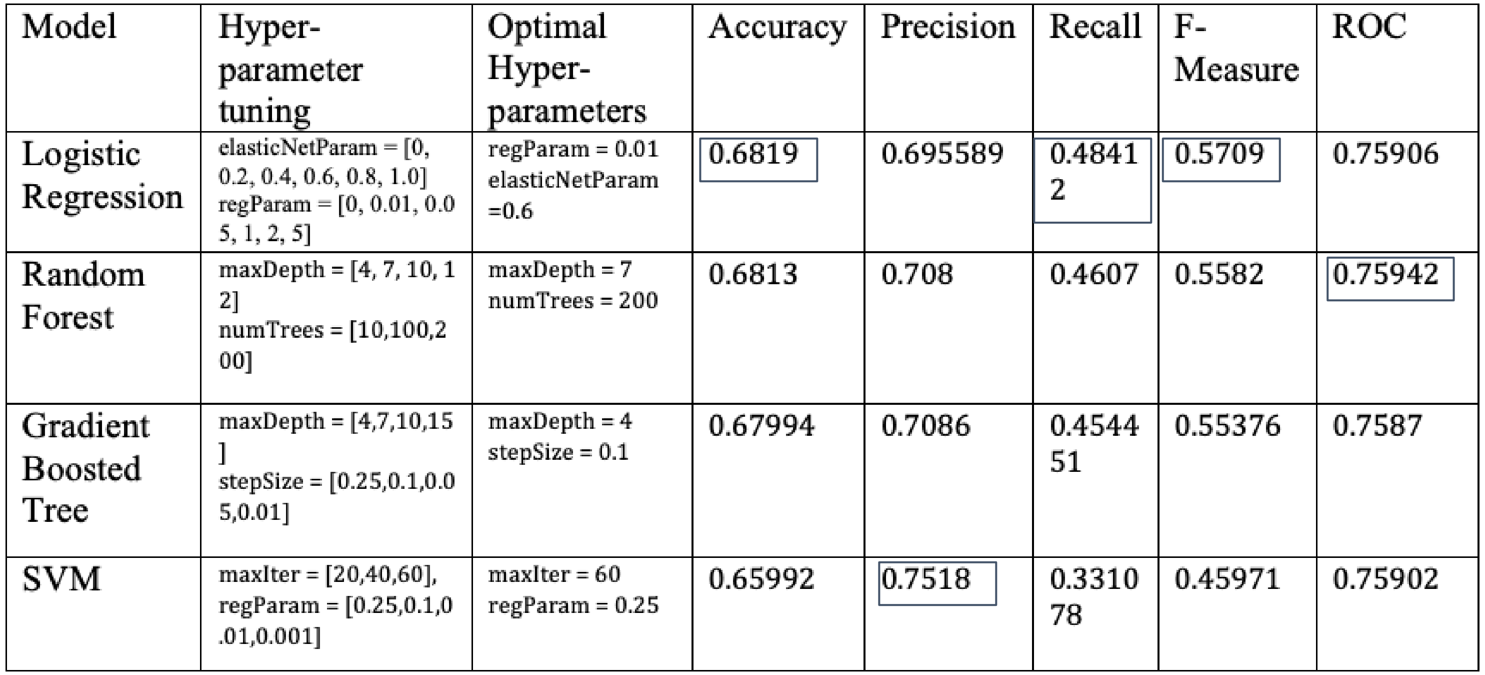


Model	Hyper-parameter tuning	Optimal Hyper-parameters	Accuracy	Precision	Recall	F-Measure	ROC
Logistic Regression	elasticNetParam = [0, 0.2, 0.4, 0.6, 0.8, 1.0] regParam = [0, 0.01, 0.05, 1, 2, 5]	regParam = 0.01 elasticNetParam = 0.2	$\boxed{0.681}$	0.6939	$\boxed{0.4826}$	0.5692	0.758
Random Forest	maxDepth = [4, 7, 10, 12] numTrees = [10,100,200]	maxDepth = 7 numTrees = 200	0.6868	0.71118	0.4768	$\boxed{0.57089}$	$\boxed{0.7581}$
Gradient Boosted Tree	maxDepth = [4,7,10,15] stepSize = [0.25,0.1,0.05,0.01]	maxDepth = 4 stepSize = 0.1	0.679	0.7086	0.4518	0.5518	0.7561
SVM	maxIter = [20,40,60], regParam = [0.25,0.1,0.01,0.001]	maxIter = 60 regParam = 0.25	0.6531	$\boxed{0.7487}$	0.3103	0.43872	0.7519

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Iteration 3: Model Tuning and Results

- Logistic regression, decision trees, random forest, gradient-boosted tree, linear SVM
- Outcome: NORM versus not NORM
- Feature selection: sex, age, bif, bif2, bpm
- Features were scaled
- Stratfolds 9 and 10 were reserved for testing based on physician's knowledge in data documentation

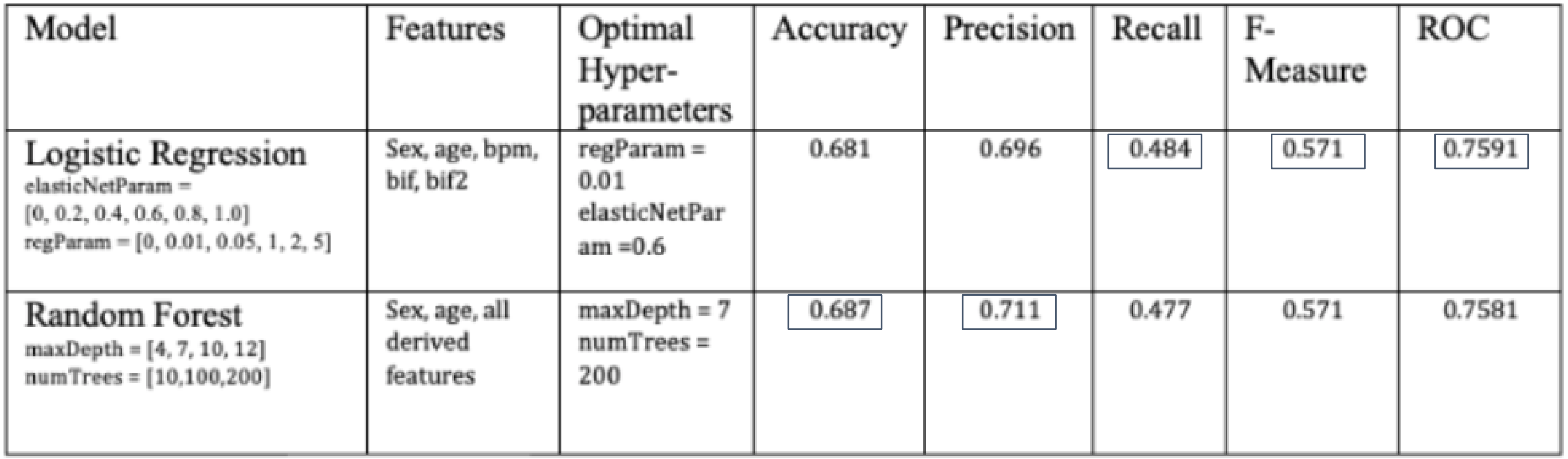


Model	Hyper-parameter tuning	Optimal Hyper-parameters	Accuracy	Precision	Recall	F-Measure	ROC
Logistic Regression	elasticNetParam = [0, 0.2, 0.4, 0.6, 0.8, 1.0] regParam = [0, 0.01, 0.05, 1, 2, 5]	regParam = 0.01 elasticNetParam =0.6	$\boxed{0.6819}$	0.695589	$\boxed{0.48412}$	$\boxed{0.5709}$	0.75906
Random Forest	maxDepth = [4, 7, 10, 12] numTrees = [10,100,200]	maxDepth = 7 numTrees = 200	0.6813	0.708	0.4607	0.5582	$\boxed{0.75942}$
Gradient Boosted Tree	maxDepth = [4,7,10,15] stepSize = [0.25,0.1,0.05,0.01]	maxDepth = 4 stepSize = 0.1	0.67994	0.7086	0.454451	0.55376	0.7587
SVM	maxIter = [20,40,60], regParam = [0.25,0.1,0.01,0.001]	maxIter = 60 regParam = 0.25	0.65992	$\boxed{0.7518}$	0.331078	0.45971	0.75902

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Optimal model selection

- By comparing the model performance metrics, we note using logistic regression with a subset of features provides optimal performance. Overall, the recall, f-measure, and ROC are higher.



Model	Features	Optimal Hyper-parameters	Accuracy	Precision	Recall	F-Measure	ROC
Logistic Regression elasticNetParam = [0, 0.2, 0.4, 0.6, 0.8, 1.0] regParam = [0, 0.01, 0.05, 1, 2, 5]	Sex, age, bpm, bif, bif2	regParam = 0.01 elasticNetParam =0.6	0.681	0.696	$\boxed{0.484}$	$\boxed{0.571}$	$\boxed{0.7591}$
Random Forest maxDepth = [4, 7, 10, 12] numTrees = [10,100,200]	Sex, age, all derived features	maxDepth = 7 numTrees = 200	$\boxed{0.687}$	$\boxed{0.711}$	0.477	0.571	0.7581

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Key Takeaways

- Although logistic regression was the optimal model, the differences in model performance metrics across all models were negligible.
- Features were created from raw data with no labelled data to verify accuracy.
- Folds 9 and 10 were the data portions used to test the data. Training was conducted on least reliable data.

UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

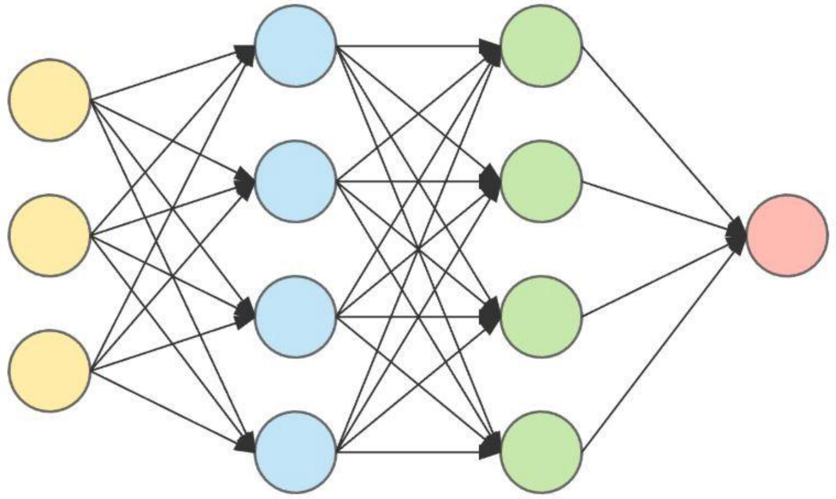
Future Work

Deep Learning and Neural Networks

- ECG feature extraction could be made easier through the use of neural networks.

- Explored the use of a neural network without much tuning. Used engineered features as input to determine whether ECG was normal or not. Also used the raw ECG data as input to see if that would improve accuracy.

- Ribeiro et al. conclude “even if a DNN is able to recognize typical ECG abnormalities, further analysis by an experienced specialist will continue to be necessary to these complex exams.”



UNIVERSITY OF VIRGINIA
DATA SCIENCE INSTITUTE

Thank You