Breast Cancer Data Analysis

PRINCESS NWABULU

MICHAEL PETERS

TOM DANNER

Executive Summary (project and project goals)

Create accurate and reliable models to predict if somebody has breast cancer and which type of breast cancer they have.

Data Sets:

Breast Cancer Wisconsin - UC Irvin:

- (https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic)

BRCA – Kaggle

- (https://www.kaggle.com/datasets/amandam1/breastcancerdataset)

BC Dataset - Selected Models

Breast Cancer Wisconsin - UC Irvine

1. Neural Network
2. XGB Classifier
3. Logistic Regression
4. Support Vector Machine

BRCA Dataset - Selected Model

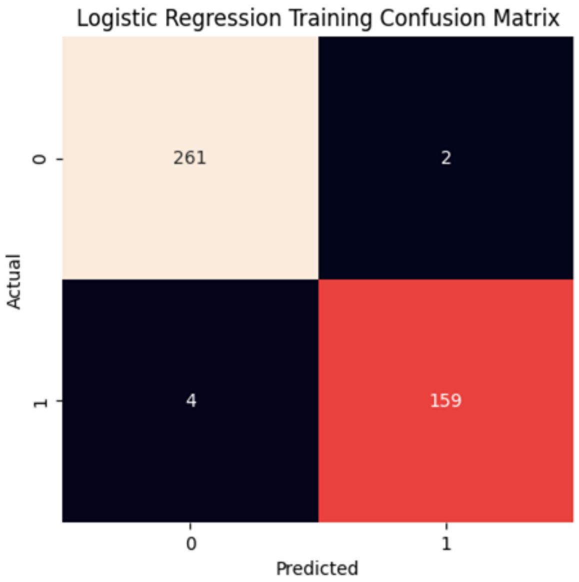
BRCA – Kaggle

1. Neural Network
2. Sequential Neural Network
3. Support Vector Machine
4. K-Nearest Neighbors- KNN

Data Preparation Process

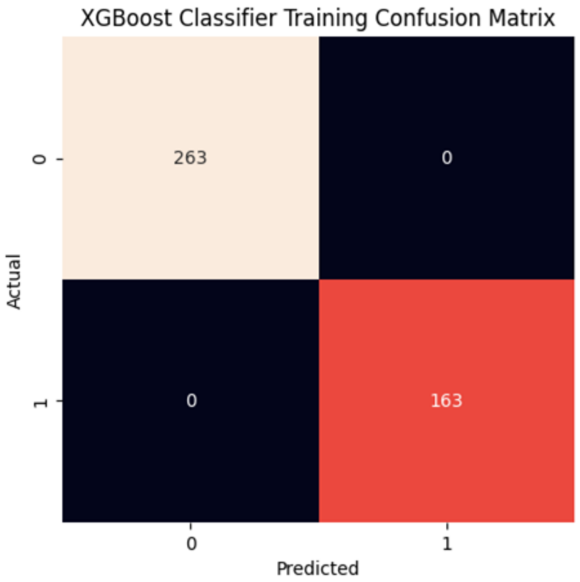
- Describe the source of your data and why you chose it for your project.
  - Kaggle is well known source for data sets.
  - we choose the BC data because of its large set of features to predict a cancer cell.
- Describe the collection, cleanup, and preparation process.
  - Collection searching on Kaggle for usable datasets
  - Identifying feature and target wanted, dropping unwanted columns, transforming data into format the models will work with using One Hot Encoder, Label Encoder, and to categorical, then Standard Scaler to scale the sets.
- Describe the training process.
  - Separating the features and target into training and testing datasets, scaling the datasets, fitting the data into the models, predicting results based on the testing data.

BC Dataset – Logistic Regression vs. XGBoost Classifier



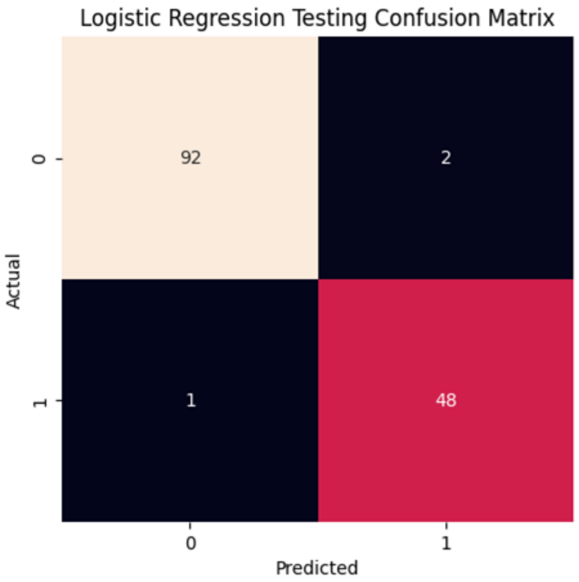
Logistic Regression Training Confusion Matrix

	0	1
0	261	2
1	4	159



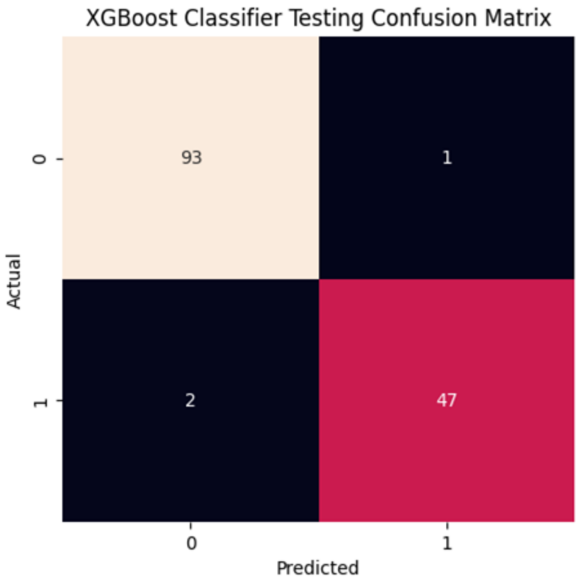
XGBoost Classifier Training Confusion Matrix

	0	1
0	263	0
1	0	163



Logistic Regression Testing Confusion Matrix

	0	1
0	92	2
1	1	48



XGBoost Classifier Testing Confusion Matrix

	0	1
0	93	1
1	2	47

BC Dataset – Classification Reports

```
Logistic Regression Model Training Classification Report
--------------------------------------------------------
              precision    recall  f1-score   support

           0       0.98      0.99      0.99       263
           1       0.99      0.98      0.98       163

    accuracy                           0.99       426
   macro avg       0.99      0.98      0.99       426
weighted avg       0.99      0.99      0.99       426

________________________________________________________

Logistic Regression Model Testing Classification Report
--------------------------------------------------------
              precision    recall  f1-score   support

           0       0.99      0.98      0.98        94
           1       0.96      0.98      0.97        49

    accuracy                           0.98       143
   macro avg       0.97      0.98      0.98       143
weighted avg       0.98      0.98      0.98       143
```

```
XGB Classifier Model Training Classification Report
--------------------------------------------------------
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       263
           1       1.00      1.00      1.00       163

    accuracy                           1.00       426
   macro avg       1.00      1.00      1.00       426
weighted avg       1.00      1.00      1.00       426

________________________________________________________

XGB Classifier Model Testing Classification Report
--------------------------------------------------------
              precision    recall  f1-score   support

           0       0.98      0.99      0.98        94
           1       0.98      0.96      0.97        49

    accuracy                           0.98       143
   macro avg       0.98      0.97      0.98       143
weighted avg       0.98      0.98      0.98       143
```

BC Dataset
Neural Network

Setup

Layers

- nn.add(Dense(units=16, input_dim=31, activation="relu"))
- nn.add(Dense(units=8, activation="relu"))
- nn.add(Dense(units=1, activation="sigmoid"))

Compilation

- nn.compile(loss="binary_crossentropy", optimizer="adam", metrics=["accuracy"])

Fitting the model

- nn.fit(X_train_scaled, y_train, epochs=200)

Results

Evaluation

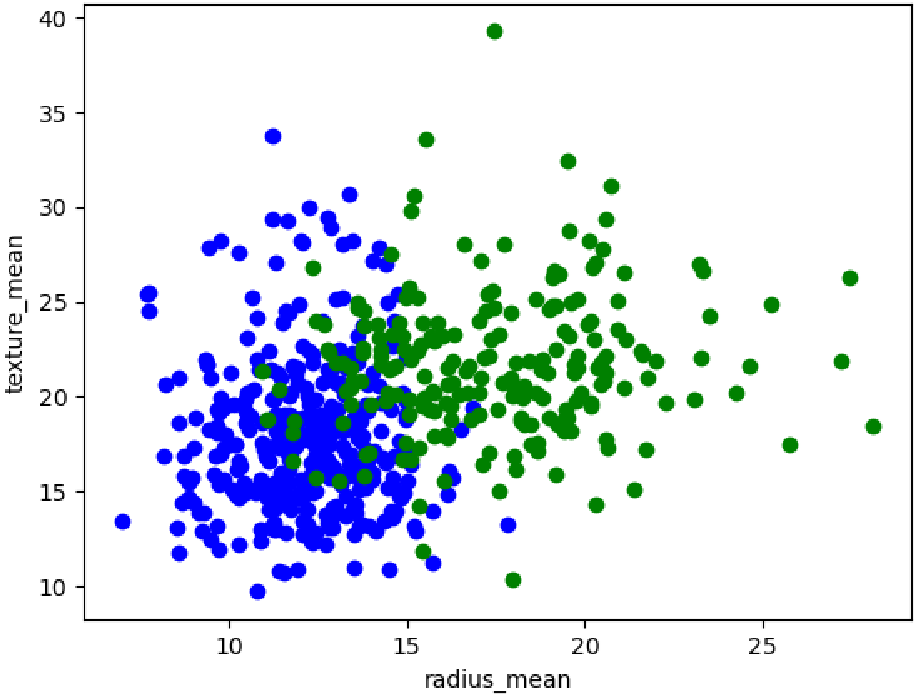
- loss: 0.1247
- categorical_accuracy: 0.9650

Predictions

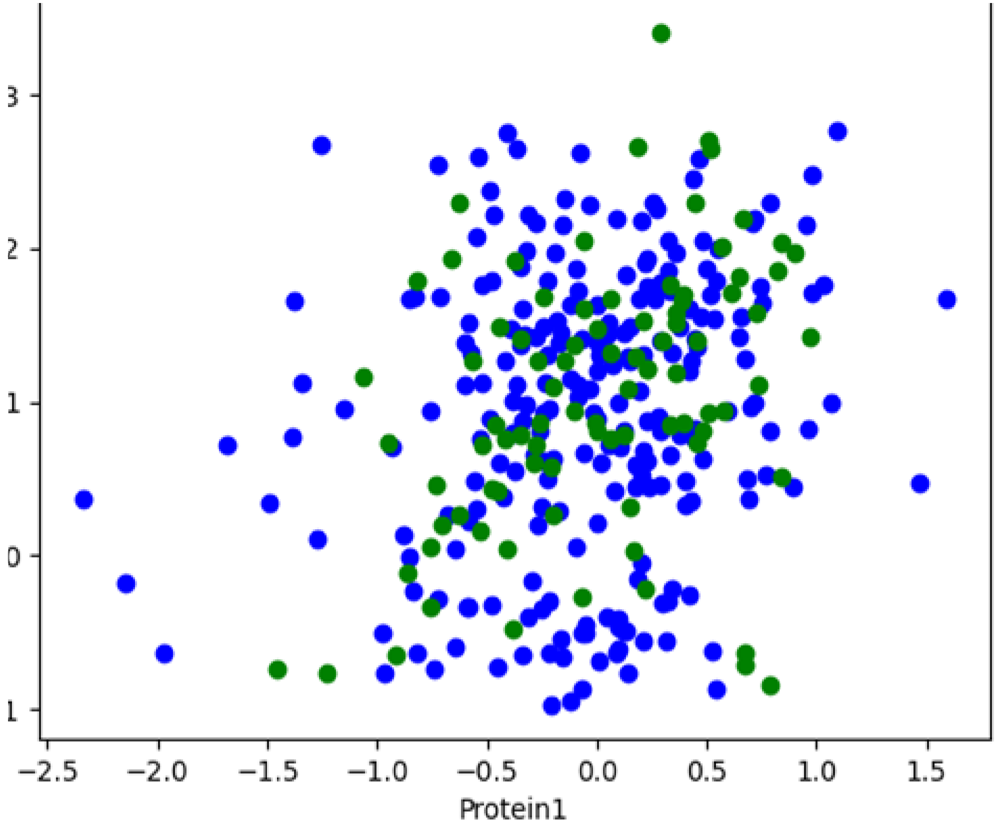
- Benign (0.0) - 88
- malignant (1.0) - 55

BC Dataset
svm

BC data



BRCA data



Data Preprocessing

BC Data

559 rows/31 columns

no nulls

y= diagnosis

x= drop diagnosis

BRCA Data

Dropped 6 columns

one hot encoder(histology)

nan after encoding

y= histology

x= drop histology

SVM

BC Data

- Kernel selection - Linear
- Parameter Tuning- N
- 80% train set
- random state=1
- Acc score = 0.96

BRCA Data

- imbalanced data?
- Parameter Tuning- N
- 80% train set
- random state=1
- Acc score = 0.58

ann/knn

BC Data

ANN

Tuning- N

BRCA Data

KNN

Tuning- y

neighbor classifier- 2,3,4

2 accuracy= 0.51

3 accuracy= 0.46

4 accuracy= 0.55

BRCA Dataset
Neural Network

Setup

Layers

- nn.add(Dense(units=7, input_dim=14, activation="relu"))
- nn.add(Dense(units=4, activation="relu"))
- nn.add(Dense(units=2, activation="relu"))
- nn.add(Dense(units=1, activation="relu"))
- nn.add(Dense(units=1, activation="relu"))

Compilation

- nn.compile(loss="binary_crossentropy", optimizer="adam", metrics=["accuracy"])

Fitting the model

- nn.fit(X_train_scaled, y_train, epochs=50)

Results

Evaluation

- loss: (NaN)
- categorical_accuracy: 0.6429

Predictions

- Infiltrating Ductal Carcinoma (0.0) - 54
- Mucinous Carcinoma (1.0) - 4
- Infiltrating Lobular Carcinoma (2.0) - 20
- No Prediction (NaN) - 6

BRCA Dataset
Sequential Neural Network

Setup

Layers

- (Dense(25, input_dim = number_of_predictors, activation = 'relu'))
- (Dense(25, activation = 'relu'))
- (Dense(25, activation = 'tanh'))
- (Dense(25, activation = 'tanh'))
- (Dropout(.1))
- (Dense(number_of_classes, activation='softmax’))

Compilation

- (loss="categorical_focal_crossentropy", optimizer= "adam", metrics=['categorical_accuracy’])

Fitting the model

- number_of_epochs = 200
- nn.fit(X_train_scaled, y_train, epochs = number_of_epochs, shuffle = True)

Results

Evaluation

- loss: 0.1370
- categorical_accuracy: 0.6875

Predictions

- Infiltrating Ductal Carcinoma - 80

Group Approach to Achieve Project Goals

Evaluation Methods

- .evaluate (Neural Networks)
- confusion matrix, classification report(Logistic Regression/XGB Classifier)

Unanticipated Problems

- NaNs in y_test (Neural Network BRCA)
- Low accuracy score for the BRCA model.
- How to display the Confusion Matrix.

Results / Conclusions

Almost perfectly predict if somebody had cancer.

- BC Dataset
  - 96.5% Accurate (Neural Network)
  - 99% training, 98% testing Accuracy (LR Model)
  - 100% training, 98% testing Accuracy (XGB Classifier Model)
  - 95.6% accurate (SVM)

Not able to predict the type of cancer based on the provided data with enough accuracy.

- BRCA Dataset
  - 68.75% Accurate (Sequential Neural Network)
  - 64.29% Accurate (Neural Network)
  - 58.20% Accurate(SVM)
  - 55% Accurate(KNN)

Next Steps

- BC Dataset
  - Find more testing data
  - Testing Accuracy >98%

- BRCA Dataset
  - Deeper analysis of model failure.
  - Try to determine which features are necessary to predict cancer type
  - Find Larger dataset for training
  - Try over sampling or SMOTE to account for imbalanced targets.