Springboard Data Science Course

Data Science Capstone Project 1

Orthopedic Biomechanical Features

Michelle Ide - 12/29/2020

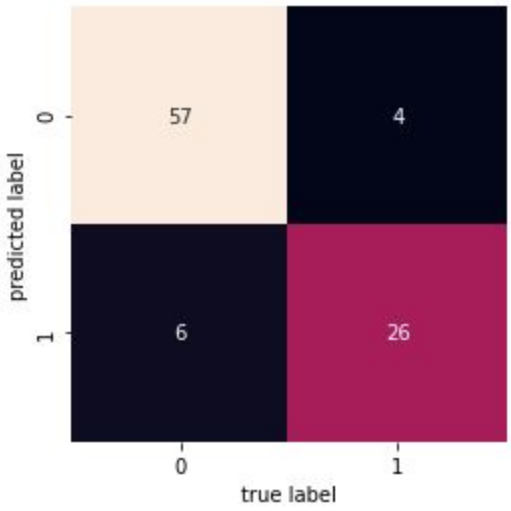
~~~ MACHINE LEARNING ~~~

Machine learning models were created and tested for this Capstone project. The project creates machine models that quickly determine if a patient is suffering from spondylolisthesis. This is determined by using 6 quantitative features containing angular orthopedic measurements and classifying each record as normal or abnormal (spondylolisthesis).

This supervised learning classification problem contains unbalanced data, ⅓ of the data is in the Abnormal range. Models were selected based on their classification strengths, parameters were turned using KFold within a GridSearch algorithm to prevent overfitting with upsampling to balance data. Resampling of the data was performed using both SMOTE and ADASYN for comparison during parameter tuning since they provide some slight changes in balance thresholds.

Below is a quick reference of the data, techniques, and accuracy results included:

- 309 Cleaned records
- Features: 6 quantitative - scaled during tuning
- Target: 1 binomial - Abnormal/Normal
- 209 Normal encoded as 0
- 100 Abnormal encoded as 1
- α of 0.5% for hypothesis test
- Stratified Test-Train split
- KFold cross validation to prevent overfitting
- Pipeline: 1) upsamples minority samples
  2) gridsearch-cross val parameter tuning
- Accuracy: F1, ROC AUC and Confusion Matrix
~~~

Models Tested: 5-Discriminative, 1-Generative

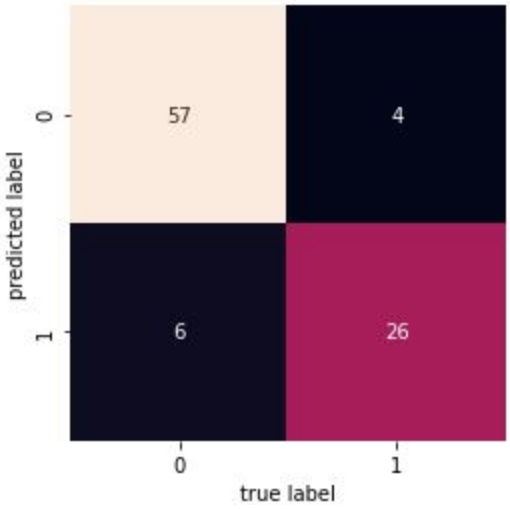
- ➢ Logistic Regression
- ➢ KNearest Neighbors
- ➢ Random Forest
- ➢ SVM
- ➢ Gradient Boost
- ➢ Naive Bayes (Generative)

Results

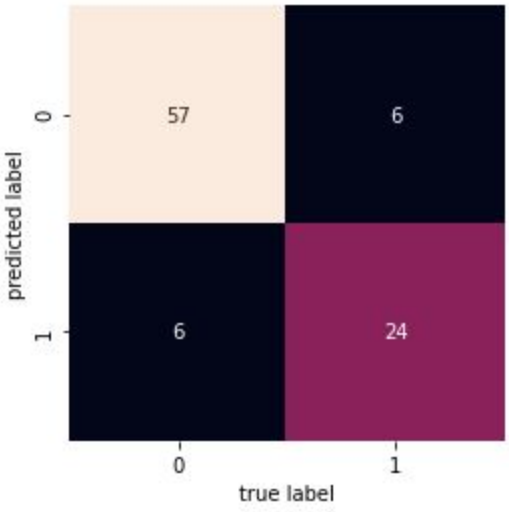
Logistic Regression



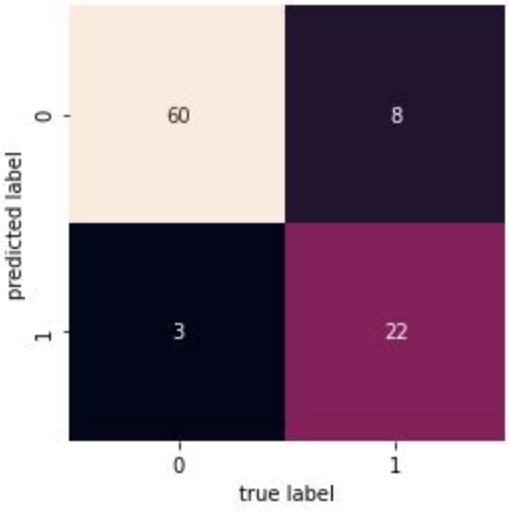
Gradient Boosting Classifier



Support Vector Machine

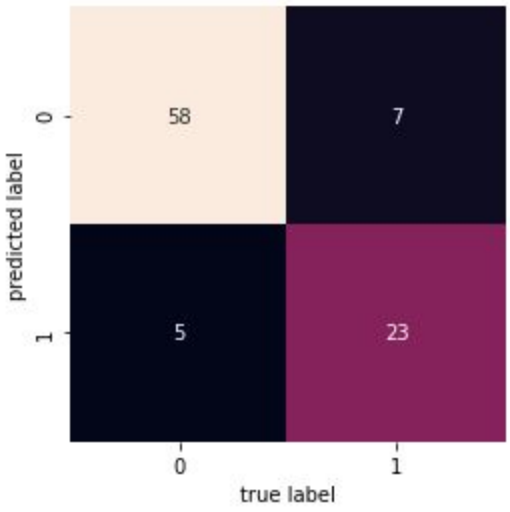


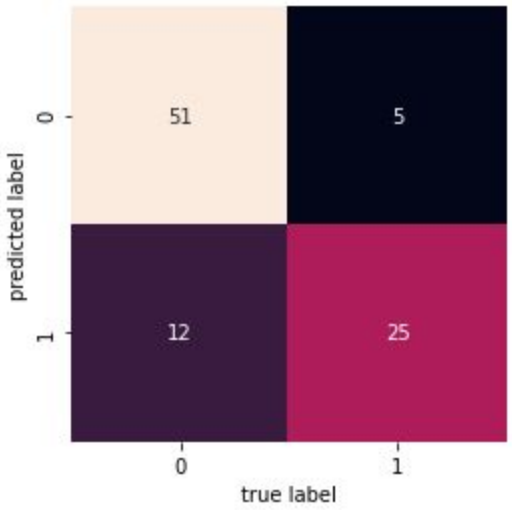
Random Forest Classifier



KNneighbors

Gaussian Naive Bayes

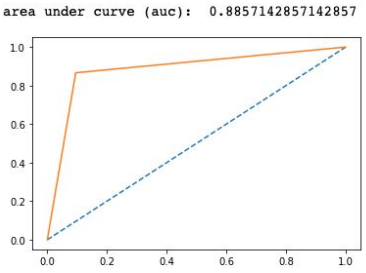




LogisticRegression(random_state=42)

	precision	recall	f1-score	support
0	0.93	0.90	0.92	63
1	0.81	0.87	0.84	30
accuracy			0.89	93
macro avg	0.87	0.89	0.88	93
weighted avg	0.90	0.89	0.89	93

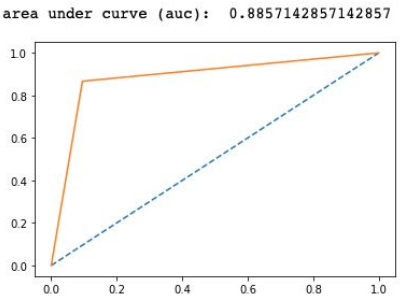
area under curve (auc): 0.8857142857142857



GradientBoostingClassifier(random_state=42)

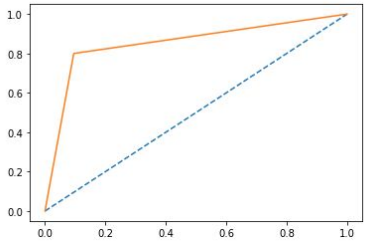
	precision	recall	f1-score	support
0	0.93	0.90	0.92	63
1	0.81	0.87	0.84	30
accuracy			0.89	93
macro avg	0.87	0.89	0.88	93
weighted avg	0.90	0.89	0.89	93

area under curve (auc): 0.8857142857142857

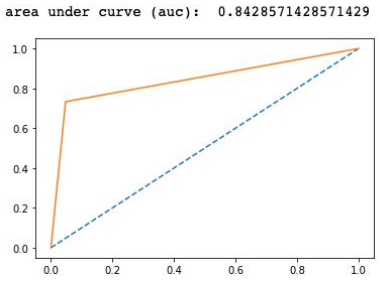


```
SVC(random_state=42)
              precision    recall  f1-score   support

           0       0.90      0.90      0.90        63
           1       0.80      0.80      0.80        30

    accuracy                           0.87        93
   macro avg       0.85      0.85      0.85        93
weighted avg       0.87      0.87      0.87        93

area under curve (auc):  0.8523809523809524
```

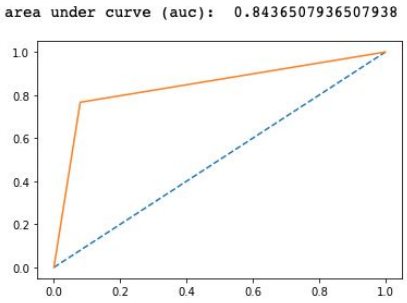
```
area under curve (auc):  0.8523809523809524
```



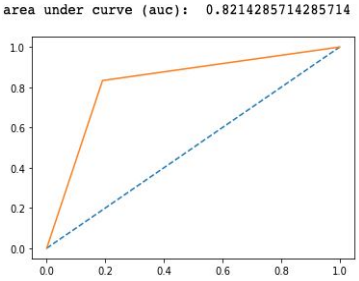
```
RandomForestClassifier(random_state=42)
              precision    recall  f1-score   support

           0       0.88      0.95      0.92        63
           1       0.88      0.73      0.80        30

    accuracy                           0.88        93
   macro avg       0.88      0.84      0.86        93
weighted avg       0.88      0.88      0.88        93

area under curve (auc):  0.8428571428571429
```




```
KNeighborsClassifier()
              precision    recall  f1-score   support

           0       0.89      0.92      0.91        63
           1       0.82      0.77      0.79        30

    accuracy                           0.87        93
   macro avg       0.86      0.84      0.85        93
weighted avg       0.87      0.87      0.87        93

area under curve (auc):  0.8436507936507938
```




```
GaussianNB()
              precision    recall  f1-score   support

           0       0.91      0.81      0.86        63
           1       0.68      0.83      0.75        30

    accuracy                           0.82        93
   macro avg       0.79      0.82      0.80        93
weighted avg       0.83      0.82      0.82        93

area under curve (auc):  0.8214285714285714
```



With parameter tuning, both Logistic Regression and Gradient Boosting algorithms performed best, and identically: ROC AUC of 88.56%, an f1 score for normal: 84% and abnormal: 92% with only 11% of test data mislabeled.

SVM, RandomForest also performed well with f1 scores at or above 80% for all and ROC AUC averaging 84%.

KNeighbors and GaussianNB broke down below 80% for normal results.