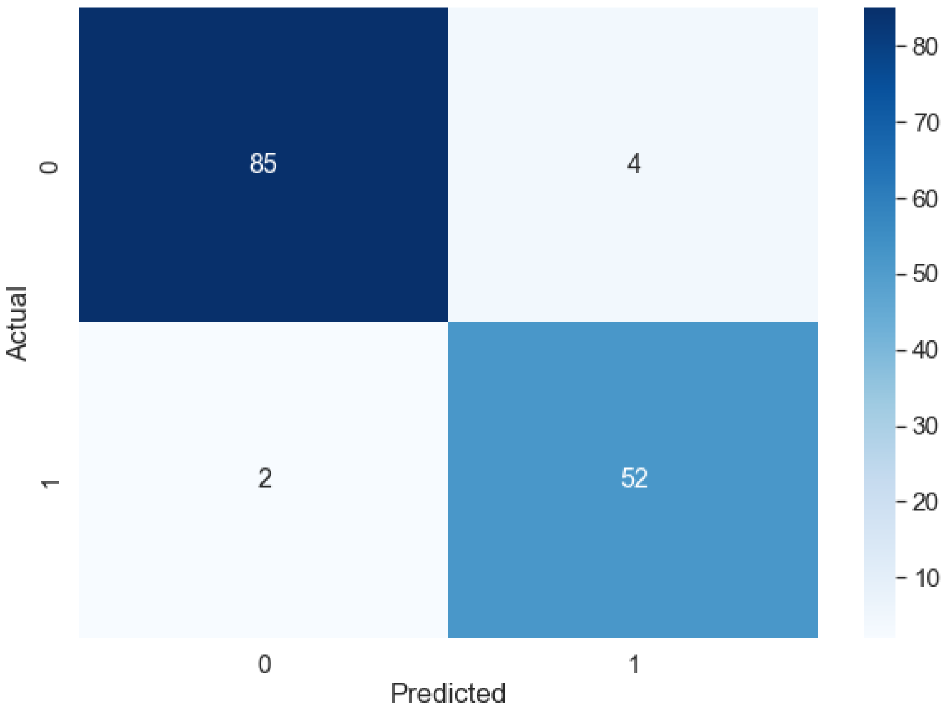
SVM (Support Vector Machines) Model

Pre-processing steps for each model:

- Split data into X = data & y = diagnosis
- Label encoded the y data to change from string (B for benign, M for malignant) to int (0 for benign, 1 for malignant)
- Scaled the X data using StandardScaler
- Split the datasets into training & testing using train_test_split with random_state = 42
- From sklearn.svm import SVC (kernel = “linear”)
- Fit the training data to the model
- Testing Data Score: **0.958**
- Made predictions for y values
- Created confusion matrix to compare predictions to actual
- Confusion matrix shows that this model tended to slightly overpredict malignancy, but still had a low number of False Negatives



Benign = 0 & Malignant = 1

The above confusion matrix visually shows how our model did:

85 True Positives = True Benign Tumors

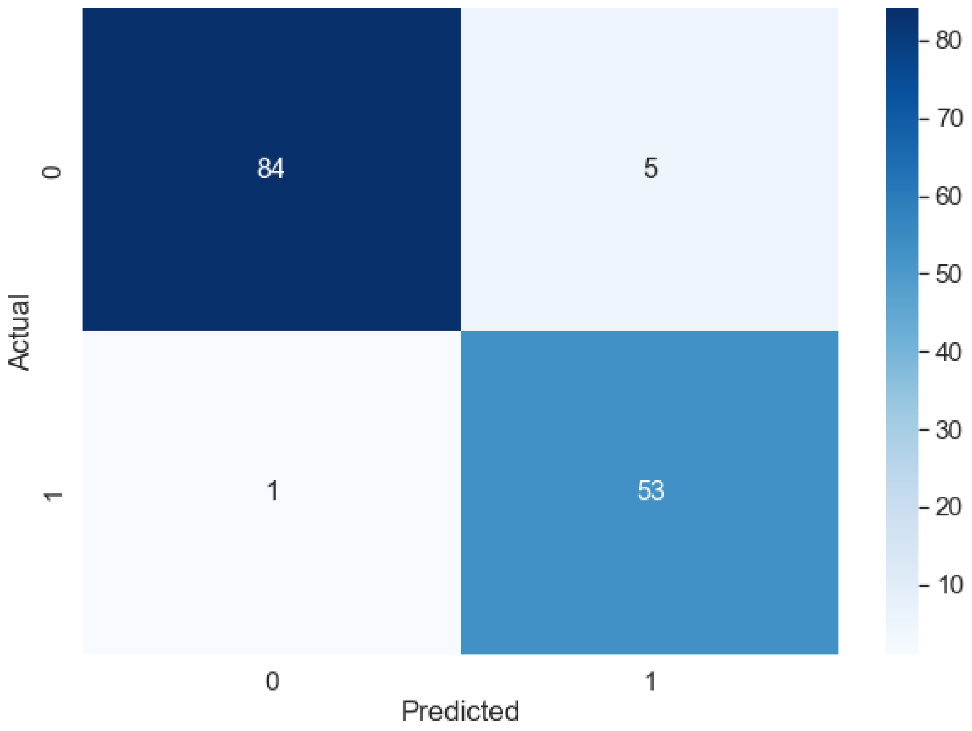
4 False Positives = Predicted to be Malignant Tumors, but actually Benign Tumors

2 False Negatives = Predicted to be Benign Tumors, but actually Malignant Tumors

52 True Negatives = True Malignant Tumors

Hypertuned SVM Model:

- From sklearn.model_selection import GridSearchCV
- Param_grid had C = 1, 5, 10, and gamma = 0.0001, 0.001, 0.01
- Best grid score = **0.979**
- Made predictions for y values, now for hypertuned model
- Created additional confusion matrix, which showed a slightly higher overprediction of malignancies, but the number of False Negatives was lower than before hypertuning



Benign = 0 & Malignant = 1

The above confusion matrix visually shows how our model did:

84 True Positives = True Benign Tumors

5 False Positives = Predicted to be Malignant Tumors, but actually Benign Tumors

1 False Negative = Predicted to be Benign Tumors, but actually Malignant Tumors

53 True Negatives = True Malignant Tumors