Capstone Project 1: Final Report

Mehmet Erdoğdu

If breast cancer is left untreated, the cancer spreads out to other parts of the body if it is a malignant cell growth. Benign cells are usually localized and do not spread to other parts. The average 5-year survival rate for women with invasive breast cancer is 91%. The average 10-year survival rate for women with invasive breast cancer is 84%. This year, an estimated 276,480 women in the United States will be diagnosed with invasive breast cancer, and It is estimated that 42,690 deaths (42,170 women and 520 men) from breast cancer will occur this year.I will predict if a given cancer cell is benign or malignant to be able to treat the cancer cells in a timely manner. The clients for this project would be hospitals and medical institutions. They care about this problem because using a highly accurate prediction model can reduce lives lost due to cancer by taking necessary preventive actions on malignant cancer cells.

The Dataset

I used the University of California, Irvine Machine Learning repository Breast cancer diagnostic data set. I acquired the data set through Kaggle website, which can be found at this link:https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

The dataset has 33 columns with 569 rows. There are ten computed real-valued features for each cell nucleus in the data set: radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, fractal dimensions. The mean, standard error and "worst" or largest (mean of the three largest values) of these features are also computed resulting in 30 features. There are a total of 357 Benign and 212 malignant points in the data set.

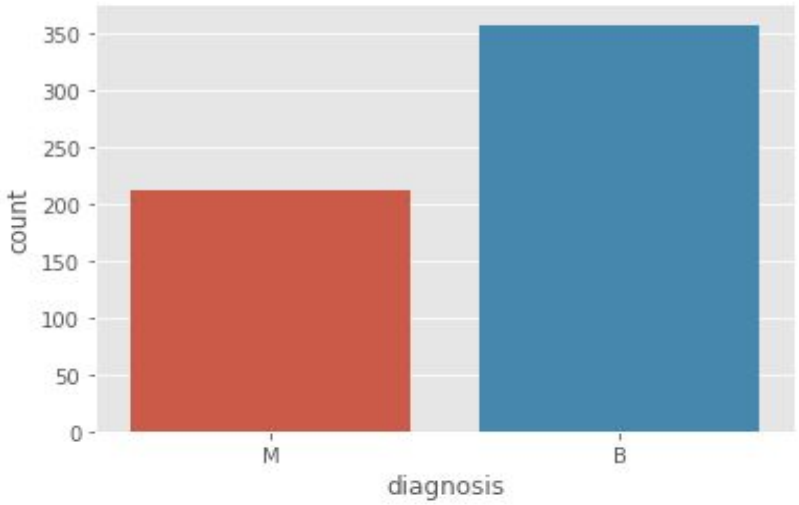


Figure 1 Benign and malignant groups

Reducing Features

As each feature has 3 corresponding columns - worst, mean, and standard error, I thought it prudent to reduce those features down to mitigate multicollinearity. As such, I used logistic regression to select the best predictor in each group of features between worst, mean, and standard error.

Then, since I didn't want to assume that there were no good features in each group other than the "best" one, I also calculated the correlation between the best feature and the other two features in the same group to confirm that multicollinearity was an issue. If the correlations were smaller than .5 between the two features I would also add it to my best features list. The final list consists of 13 features: radius worst, texture worst, perimeter worst, area worst, smoothness worst, compactness mean, concavity mean, concave points worst, symmetry worst, fractal dimension worst. radius worst, texture worst, perimeter worst, area worst, smoothness worst, compactness mean, concavity mean, concave points worst, symmetry worst, fractal dimension worst.

Exploratory Data Analysis

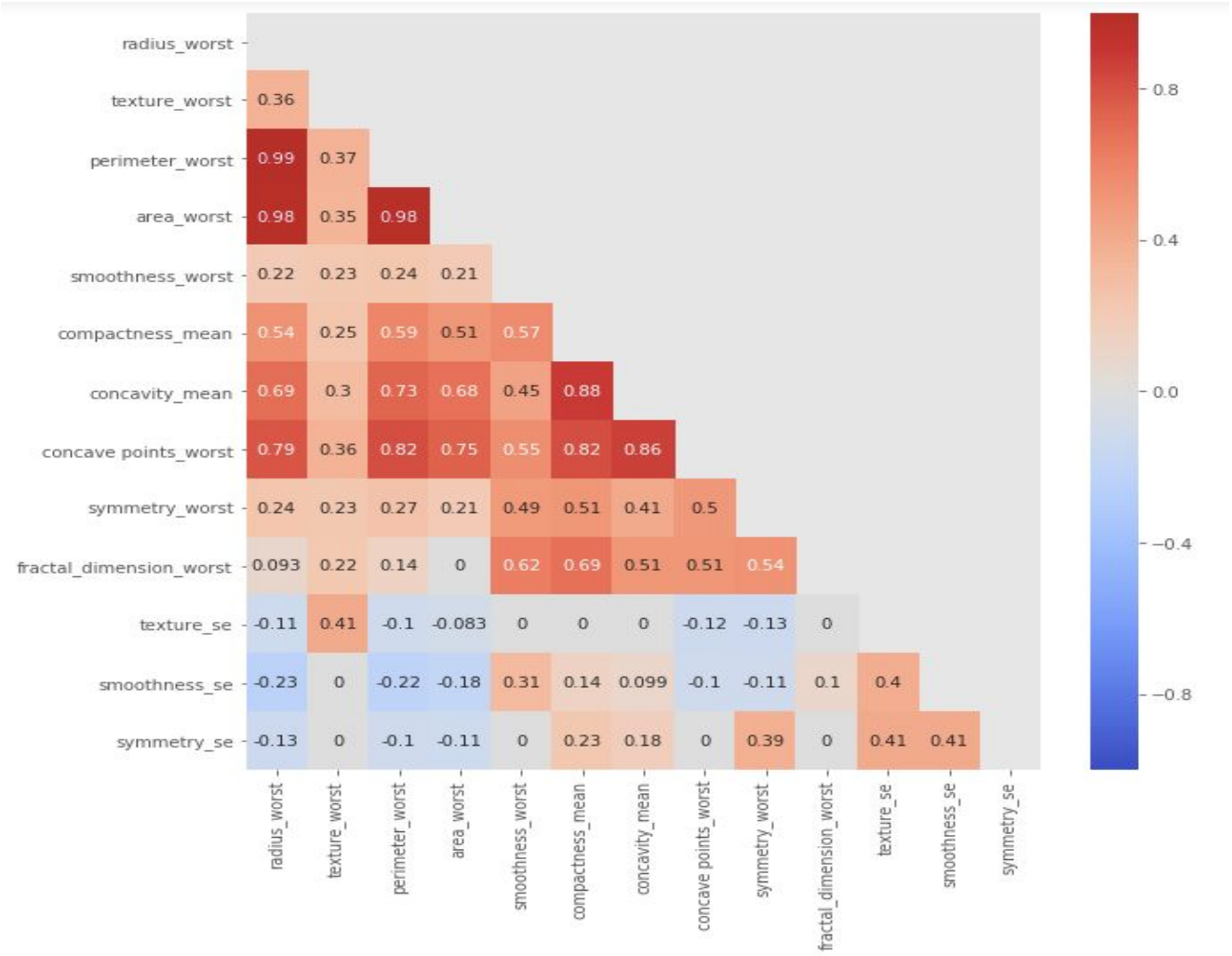


Figure 2 Correlation matrix of the best features

Above, you can see the correlation matrix of the best features. I have grayed out non-significant correlations between each two features group combinations. Correlation matrix shows that perimeter_worst,area_worst and radius_worst is highly correlated with .99, it is concerning. Also concavity_mean is highly correlated with concave_points_worst and compactness_worst.

Comparing Malignant vs Benign

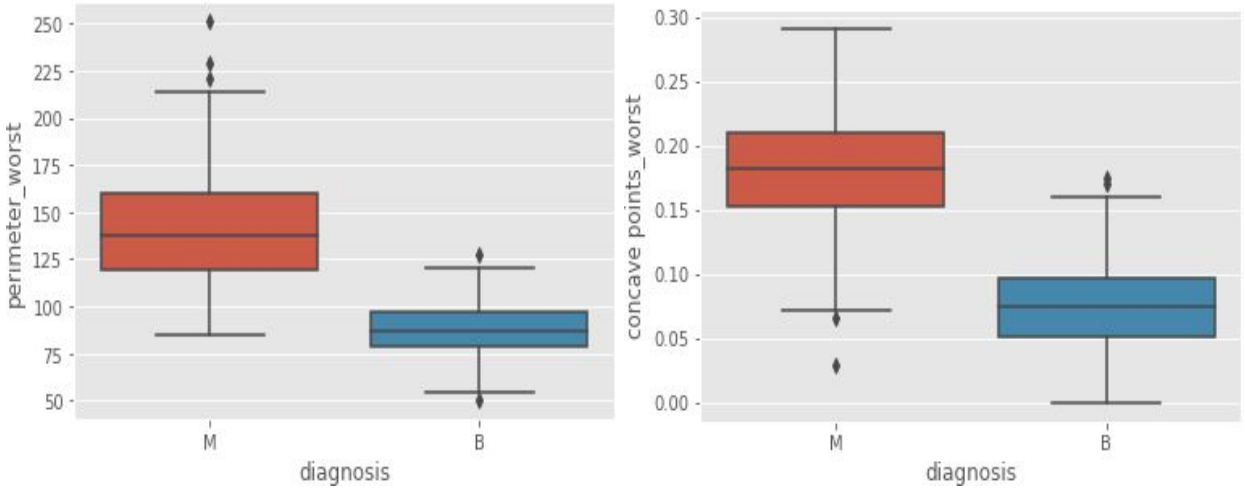


Figure 3 Perimeter and Concave points box plots

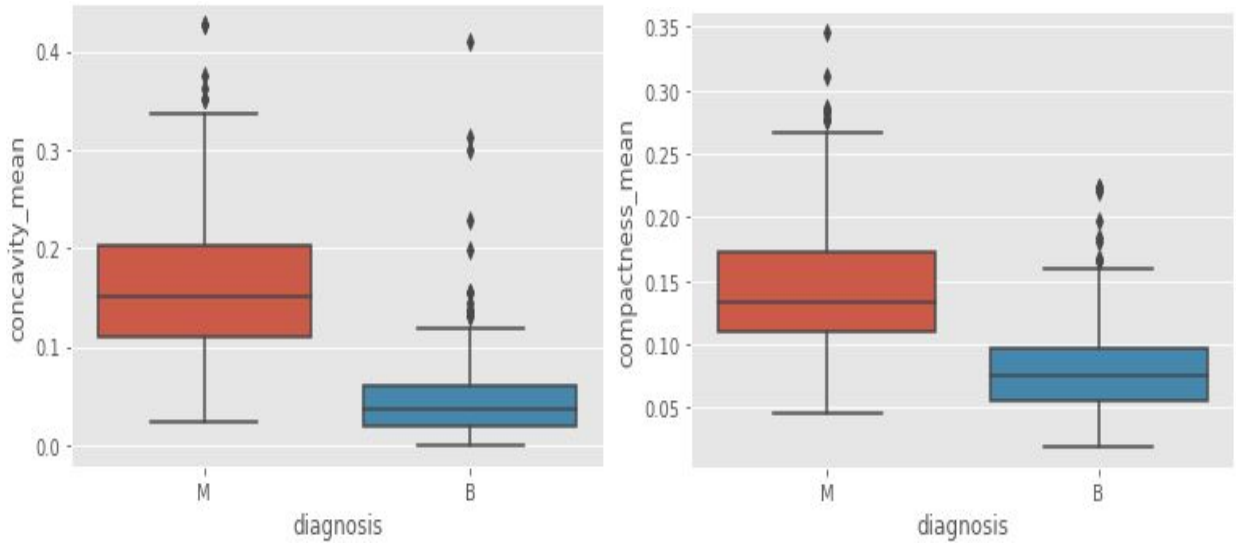


Figure 4 Concavity and compactness box plots

As the boxplots above show, Malignant and Benign cells tend to have different characteristics which makes it seem promising that they can be distinguished with a predictive model. In particular, concavity, perimeter, compactness, and number of concave points are all larger in Malignant cells than in Benign cells. Statistical testing using a t-test confirms that this relationship is statistically significant across all of these features with $p < .05$.

Investigating Effect of Features on Malignancy

I started investigating the effect of features by detecting multicollinearity with Variance Inflation Factor(VIF). To be able to calculate VIF scores, I needed to standardize all the features using StandardScaler. As we can see on the VIF table on the left below, radius_worst,

perimeter_worst and area_worst have high variation inflation factor because they explain same variance.I drop the two features that has the highest two variance.Also,concavity_mean and concave points_mean are highly correlated and they have high vif so I drop the concave points_mean as well.

	variables	VIF
0	radius_worst	154.287634
1	texture_worst	2.389733
2	perimeter_worst	148.501728
3	area_worst	39.614820
4	smoothness_worst	3.429665
5	compactness_worst	9.557115
6	concavity_mean	11.482680
7	concave points_mean	17.139147
8	symmetry_worst	3.896686
9	fractal_dimension_worst	5.453913
10	texture_se	2.690228
11	smoothness_se	2.610331
12	symmetry_se	3.429996

	variables	VIF
0	radius_worst	3.623543
1	texture_worst	2.251366
2	smoothness_worst	2.973325
3	compactness_worst	7.466057
4	concavity_mean	4.989982
5	symmetry_worst	3.858735
6	fractal_dimension_worst	4.961288
7	texture_se	2.614840
8	smoothness_se	2.553547
9	symmetry_se	3.402587

After dropping area_worst,perimeter_worst and concave points_mean features from the data set, VIF scores decreased notably.Above on the right can be seen in the VIF table without those features.

After narrowing down the features using VIF scores, I then run them through a statsmodels logistic regression.Below we can see the summary report of our logistic regression.

Dep. Variable:	dfnew.M	**No. Observations:**	569
Model:	GLM	**Df Residuals:**	558
Model Family:	Binomial	**Df Model:**	10
Link Function:	logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	nan
Date:	Tue, 28 Apr 2020	**Deviance:**	nan
Time:	08:31:55	**Pearson chi2:**	1.13e+04
No. Iterations:	100		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.5290	0.360	-1.470	0.141	-1.234	0.176
radius_worst	7.2044	1.247	5.779	0.000	4.761	9.648
texture_worst	1.7992	0.470	3.826	0.000	0.878	2.721
smoothness_worst	1.5390	0.612	2.513	0.012	0.339	2.739
compactness_worst	-0.9534	0.852	-1.120	0.263	-2.623	0.716
concavity_mean	2.4454	0.752	3.253	0.001	0.972	3.919
symmetry_worst	1.0986	0.722	1.521	0.128	-0.317	2.514
fractal_dimension_worst	-0.2440	0.790	-0.309	0.758	-1.793	1.305
texture_se	-0.1599	0.543	-0.294	0.769	-1.225	0.905
smoothness_se	0.1426	0.635	0.225	0.822	-1.102	1.388
symmetry_se	-0.4523	0.769	-0.588	0.556	-1.959	1.054

Based on this analysis, it seems clear that radius is the most important feature with a coefficient more than 3x greater than the next most important feature, concavity, when everything is scaled. Looking at the p-values, the variables 'radius_worst','texture_worst'', 'smoothness_worst', and ‘concavity_mean’ ’ seem to be the only significant predictors since their p-values are smaller than 0.05. All the other variables have their p-values larger than 0.05, and are, therefore, not significant.

The fitted model says that one unit increase in the radius_worst increases having a malignant cell by %133800 then having bening cell holding other features constant since exp(7.2) is 133.9. The coefficient for concavity_mean says that, holding other features constant,we see %1000 increases in the odds of having a malignant class for one-unit increase in concavity_mean since exp(2.44) is 11.

Machine Learning

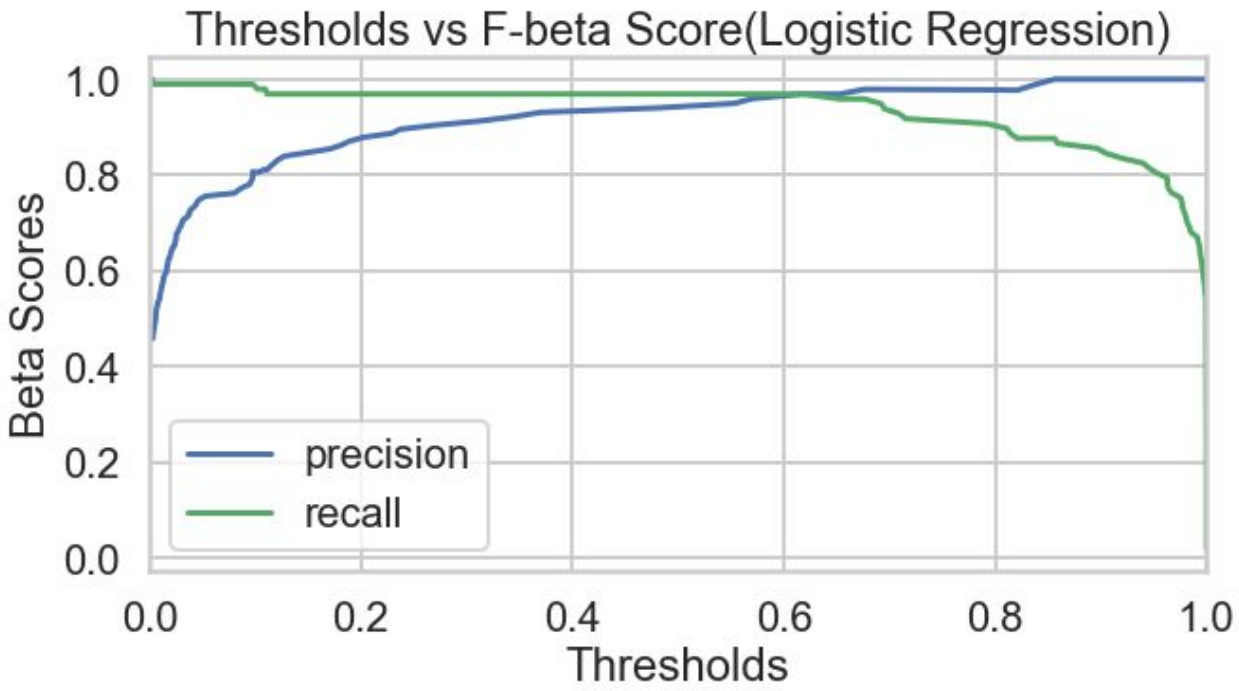
Machine Learning was performed on the dataset including only the best features as determined in the prior section. The training data was 55% of the data. The remaining 45% was my hold-out set that was left untouched until the final model has been selected and tuned. Before using different supervised machine learning approaches I performed hyperparameter tuning to find out what estimators worked most effectively. I used roc_auc as our metric to determine the best parameters for each model, grid searching over 5-fold cross-validation. The resulting best parameters for each model are below:

Model	Parameters	Roc-Auc Score
Logistic Regression	C : 31	0.988
KNN	N_neighbors : 39	0.970
Random Forest	Max_depth : 19 Max_features : 4 N_estimators : 118	0.986

Picking the Optimal Threshold

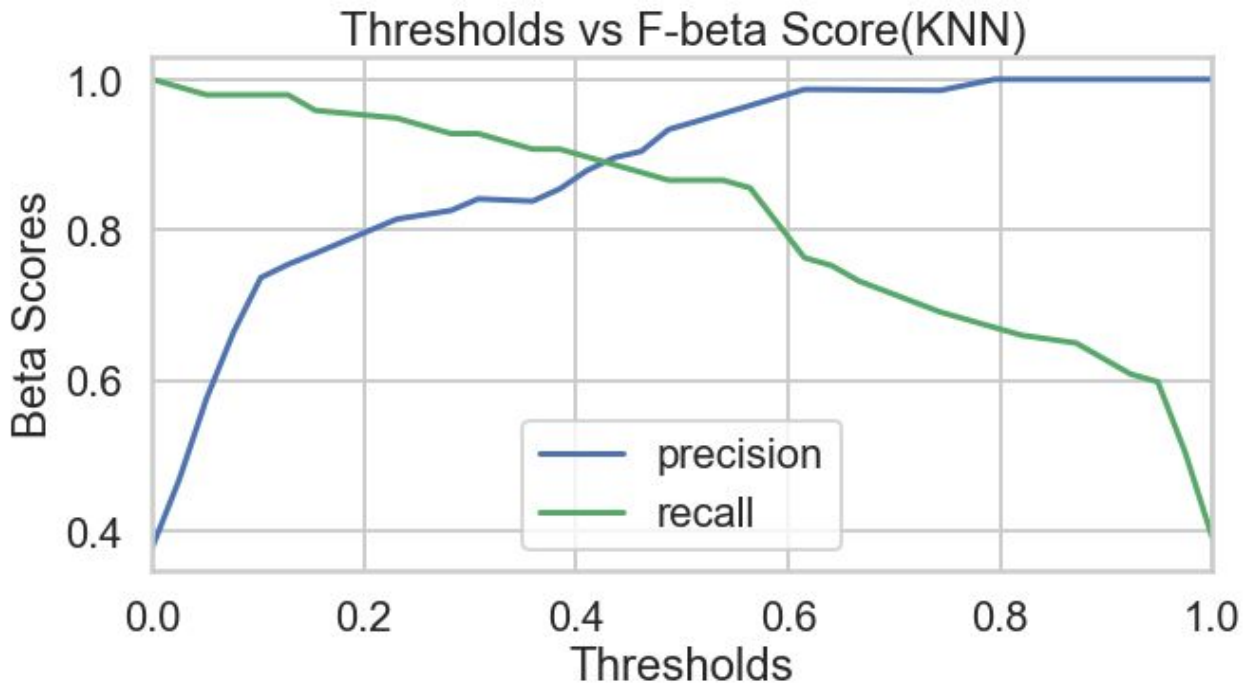
Since this is a cancer diagnostic project, it is much more important to reduce false negatives, so when a person has cancer they can start treatment as early as possible, than it is to ensure that we minimize false positives. For that reason maximizing recall is more important than precision, although precision is still important. For this reason, I chose to optimize my models for a f-beta score where 1.5 was set as the beta. Unlike in an F1-score, where recall and precision are equally weighted, by choosing a beta of 1.5 in this case I was able to weigh the recall more heavily.

After hyperparameter tuning, Ifit my model on the train set and predicted probabilities using the test set and then calculated different thresholds for our model. I then calculated precision, recall, and F1.5 scores by looping over the different thresholds. Below graphs show all the scores and thresholds and the vertical line indicates the best score and best threshold.



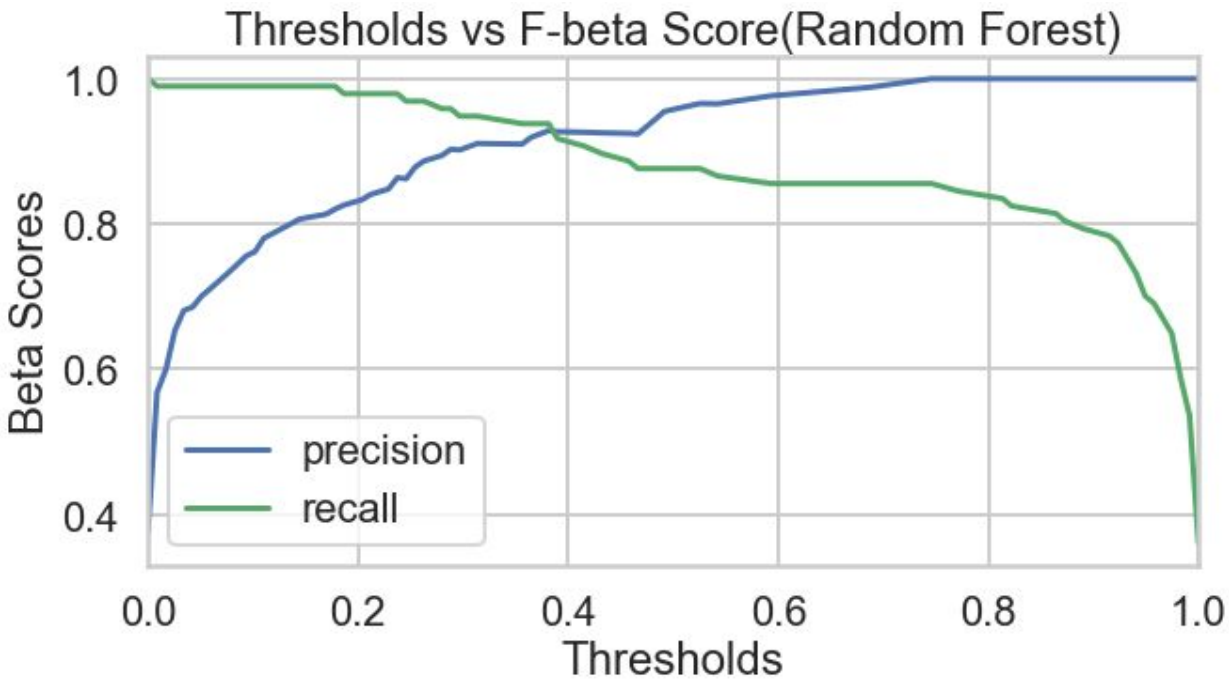
Logistic Regression Classification Report

	precision	recall	f1-score	support
0	0.98	0.97	0.97	160
1	0.95	0.97	0.96	97
accuracy			0.97	257
macro avg	0.97	0.97	0.97	257
weighted avg	0.97	0.97	0.97	257



KNeighborsClassifier Classification Report

	precision	recall	f1-score	support
0	0.92	0.97	0.95	160
1	0.94	0.87	0.90	97
accuracy			0.93	257
macro avg	0.93	0.92	0.92	257
weighted avg	0.93	0.93	0.93	257



Randon Forest Classifier Classification Report

	precision	recall	f1-score	support
0	0.92	0.97	0.95	160
1	0.94	0.87	0.90	97
accuracy			0.93	257
macro avg	0.93	0.92	0.92	257
weighted avg	0.93	0.93	0.93	257

Results

After building and testing over 3 models, I concluded that the best one is Logistic Regression Classifier with tuned hyperparameters. Using this model, I'm able to identify malignant tumors 97% of the time with only 5% false positives. Given the low false positive rate, it may even be beneficial to increase beta so as to reduce even further our 3% false negative rate.

Conclusion and Further Thoughts

Highly accurate cancer prediction can save lives. In this project, I tried to show the importance of feature selection and data visualization and model selection. Default data includes 33 features but after feature selection we drop this number from 33 to 13. My model has accurately labeled 97% of the test data. We could try to increase the True Positive rates even higher by using different algorithms or increase the beta.