SVM, LOGISTIC REGRESSION AND RANDOM FOREST FOR CANCER CLASSIFICATION

Xinyang Liu, Yifan Bian, Yichen Guo

INTRODUCTION

Different machine learning algorithms have different efficiency and accuracy when predicting an outcome. In our presentation, we will compare 5 different models when assigning them the same task and record their Accuracy, Sensitivity, Specificity, Precision, and F-measure.

In the study of Boston Breast Cancer, we want to construct a model to predict the class of tumor, benign or malignant. Ten real-value features are considered in this study and all featured values are recorded with four significant digits. Our main point is to test the advantage PCA can bring to an algorithm, especially accuracy wise. We will perform 5 different models:
Logistic regression with hyper tuning, Logistic regression with PCA,Random Forest,Random Forest with PCA and Svm and compare the results.

DATA

We utilized the University of California Irvine Machine Learning Breast Cancer data as our raw data. This data has 569 observations where each observation represents a feature. To perform a class of tumor prediction, we select 10 features and train the dataset with 5 different models. We observe that all ten features are essential for us to predict the class of tumor. With the large number of features (10) we include in our training process, we can make sure our comparison of 5 models are reasonable.

Before training our data, we perform an outlier detection and feature correlation test to clean our data.

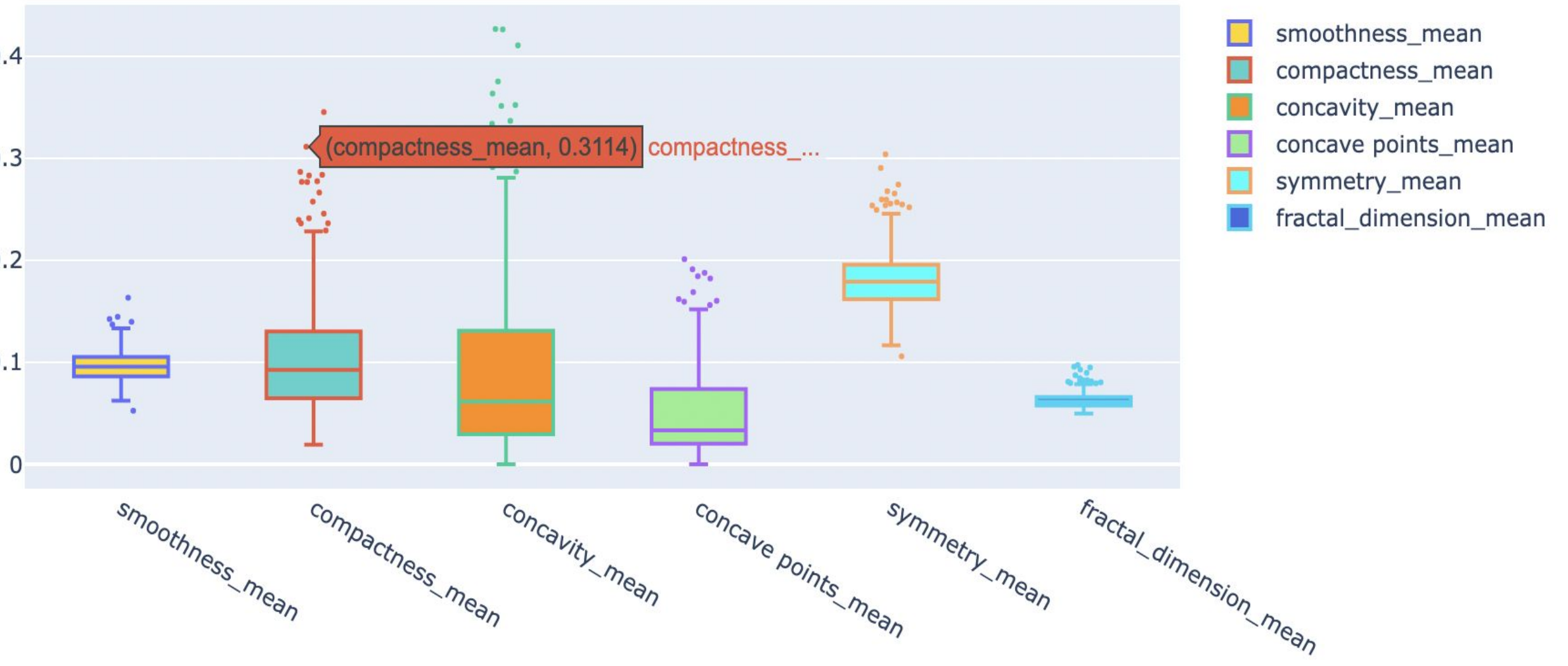

Figure 1: The boxplot of six features of the dataset (Outlier detection)

We can see that there are no outliers within our data so no further cleaning needs to be done.

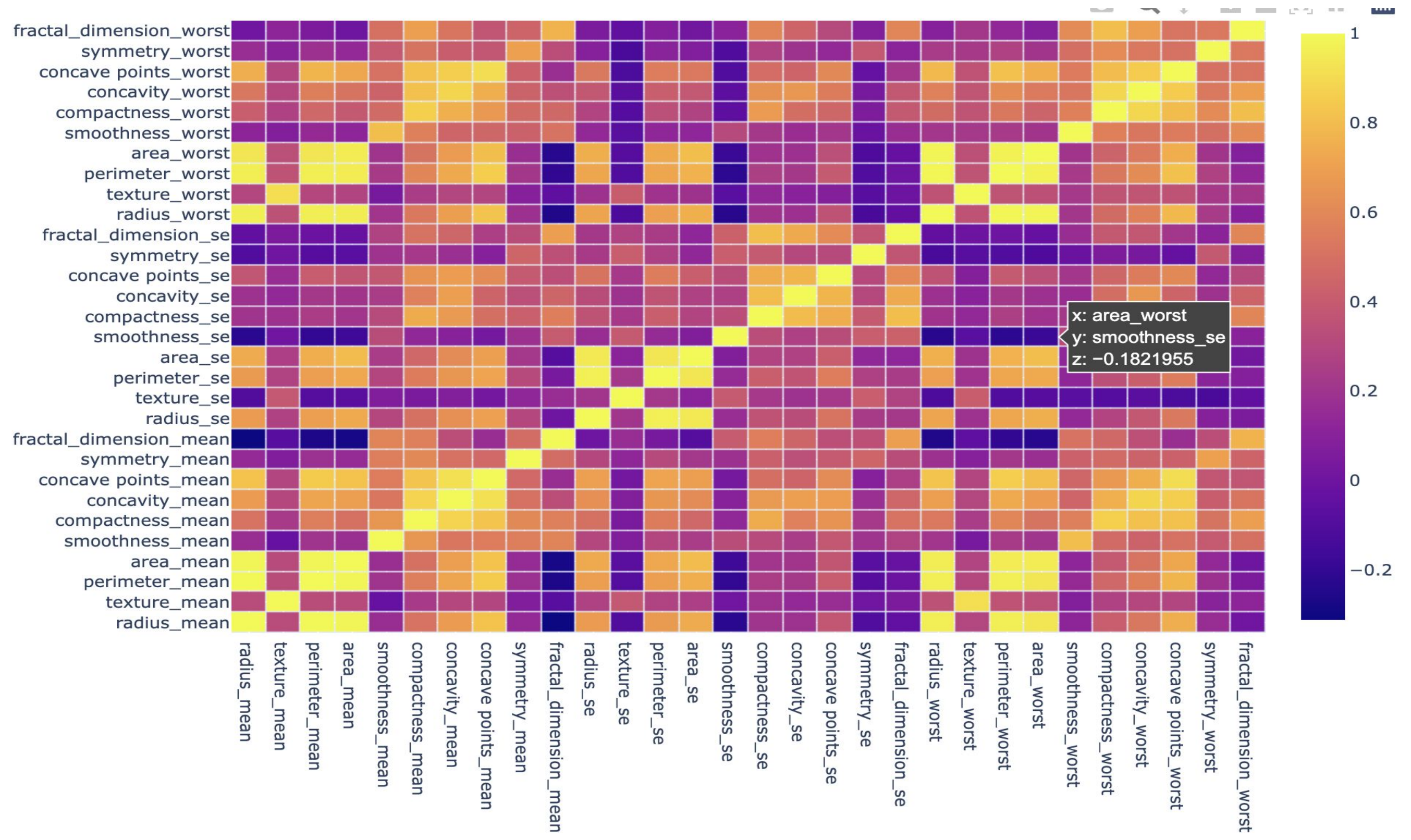

Figure 2: Correlation Matrix of all features of our dataset

Although we can see that some features are highly correlated with each other, such as area radius and perimeter, we still include all 10 features in our dataset since we don't want to lose any information and it is better include more dimensions to test the advantages of PCA.

All 10 of the features are recorded as a four digit value. We split our data into 80% of training set and 20% of testing set.

METHODOLOGY

Besides the usual procedures like splitting the cleaned and labeled dataset into training and testing set to avoid model overfitting. In order to find the best hyperparameters for each model we are testing, we decided to use ***Grid Search Cross Validation*** to find the most optimal parameters. For our project, we use scikit-learn's GridSearchCV that runs through all the different parameters that are fed into the parameter grid and produce the best combination of parameters based on accuracy metric.

The reason we choose these models is because their volatility differs a lot, some are sensitive to imbalance dataset and some are not. In real-world problems, classification algorithms including too many features in a dataset often means more data, bulky computation, and also overfitting risk. Thus we want to see the effectiveness of adding PCA to these classification algorithms.

From the correlation matrix, we observe that some features are highly correlated which means they have similar effect on the dependent variable. Both feature selection (selecting specific features to include) and feature extraction(extracting a new feature set from the input features) can effectively avoid these problems. In practice, the effectiveness of dimensionality reduction highly depends on the algorithm to be applied later and type of the data to be passed into the methodology. Here, we will compare the effectiveness by separately using Principal component analysis (PCA) on different classification algorithms.

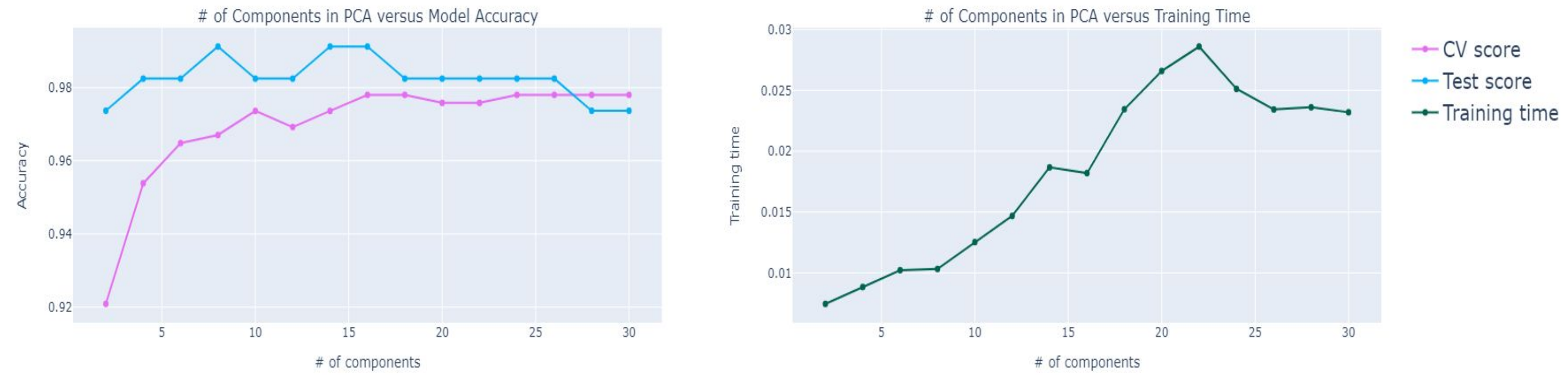

Figure 3: The relationship between number of PCA components with model accuracy and running time

To decide the number of PCA components to use in the model, we looped through a list of possible components all the way till the maximum number of features the training set has and decided on the PCA components with the highest test score; In the meantime, we will compare the computation time of calculation and try to find the optimal number of PCA components with the highest accuracy score. To compare and evaluate the performance of each algorithm, we develop five metrics (accuracy, sensitivity, specificity, precision, and F-measure) based on the confusion matrices of each model and we plot the final result together to compare across each model.

RESULTS

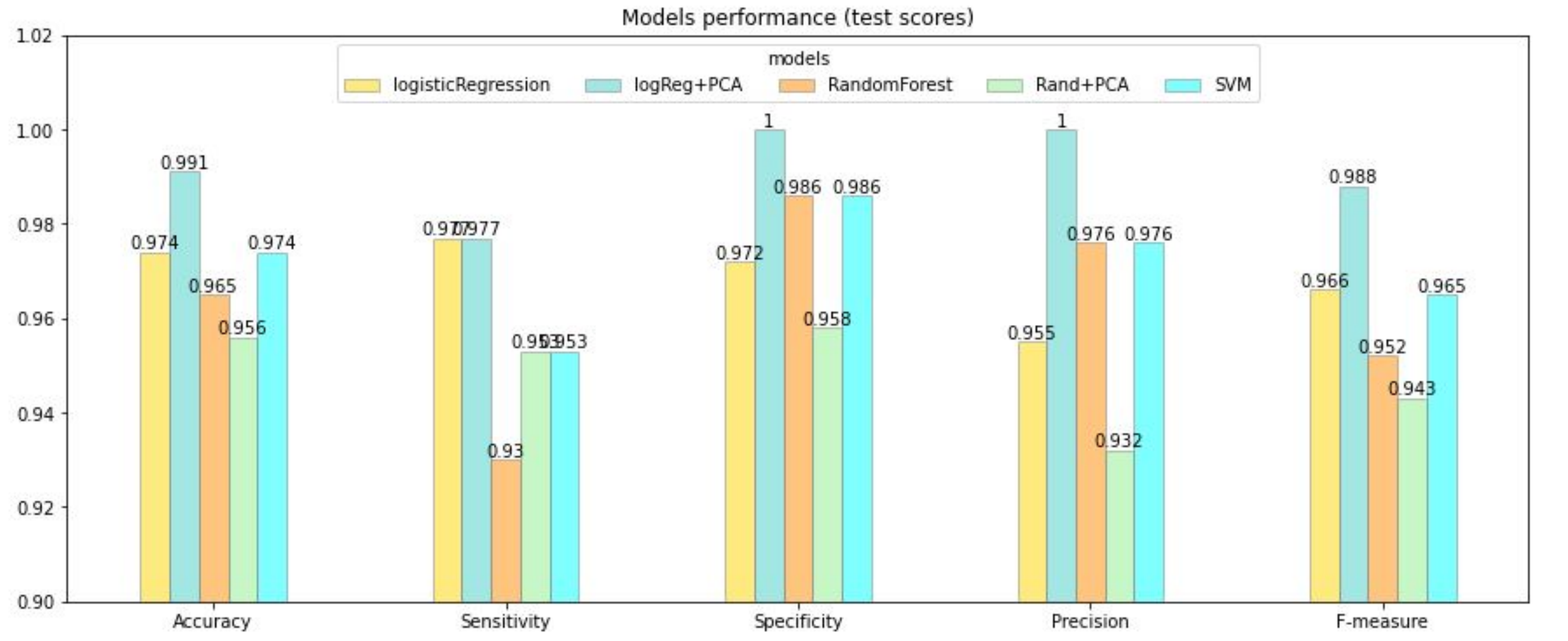

Figure 4: Comparison of each model's performance

By applying model selection and PCA component selection, we are able to found out the optimal parameters each models. For example, for logistic regression with PCA, the optimal total number of components in PCA is 8 with accuracy score of 0.991; where the best parameters for random forest classifier with PCA is {'bootstrap': False, 'criterion': 'entropy', 'n_estimators': 100}, which has a score of 0.956; The best parameters for SVM classifier is {'C': 0.4, 'kernel': 'linear'} with a training score of 0.978.

In this breast cancer dataset, the classification performance in logistic regression method is obviously improved after using PCA. In the meanwhile, we also noticed that when we use random forest technique as a classifier, the classification metrics are no overall improvement with PCA method.

DISCUSSION - THE OPTIMIZATION DETAILS

PCA:

To solve PCA problem, sklearn mainly conduct 3 steps: (1) calculate the covariance matrix of the samples.(2)find the first k largest eigenvalues and corresponding eigenvectors (3)Project all data points on the eigenvectors.

Logistic Regression:

Instead of using gradient descent or Newton's method, Sklearn applies quasi-Newton method to find the likelihood estimates in the cost function. Compared to Newton method, quasi-Newton method uses a positive definite matrix to approximate the inverse of the hessian matrix so that it can get the acceleration rate that Newton method promises as well as mitigate the cost of it. L-BFGS algorithm is used to update that term iteratively in our case.

$$x_{k+1} = x_k - \lambda_k \underline{H_k}^{-1} g_k$$

(1)The underlined part is where quasi-Newton method modifies

Random forest:

The method we use is entropy. The weaker learners consistently try all the possible thresholds and choose the one that creates the largest information gain in the next layer.

$$\mathrm{Ent}(D) = -\sum_{k=1}^{|y|} p_k \log_2 p_k$$

(2)The formula calculating entropy

$$\mathrm{Gain\_ratio}(D,a) = \frac{\mathrm{Gain}(D,a)}{\mathrm{IV}(a)}$$

$$\mathrm{IV}(a) = -\sum_{v=1}^{V} \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

(3)The formula of gain ratio

SVM:

Sklearn uses a algorithm called SMO (Sequential Minimal Optimization) to solve the cost function of the non-linear separate problem.

$$arg\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle$$

$$\text{subject to } \alpha_i \geq 0 \text{ , } \sum_{i=1}^{N} \alpha_i y_i = 0$$

(4)The original cost function of kernel SVM

SMO first selects two variables, fixing other parameters,which means that we can neglect all other parameters and the result is an equal optimization function. After finding the maximum of these two variables. it will fix them. and then select other two variables until the target value converges.

$$W(\alpha_1,\alpha_2) = \alpha_1 + \alpha_2 - \tfrac{1}{2} K_{1,1} y_1^2 \alpha_1^2 - \tfrac{1}{2} K_{2,2} y_2^2 \alpha_2^2 - K_{1,2} y_1 y_2 \alpha_1 \alpha_2 - y_1 \alpha_1$$
$$\sum_{i=3}^{N} \alpha_i y_i K_{i,1} - y_2 \alpha_2 \sum_{i=3}^{N} \alpha_i y_i K_{i,2} + C$$

(5)The cost function transformed by SMO

To find the maximum of every single pair, there is one outer loop and an inner loop. The outer loop try to find out the sample which violates the KKT conditions most seriously. The inner loop try to search the one that can minimize the optimized objective function value

CONCLUSION

It turns out that a combination of logistic regression and PCA decomposition lead to the best prediction. Considering that the performance of SVM is also overall acceptable, the reason may be that the task we faced here is a linear separate problem.

The effectiveness of dimensionality reduction highly depends on the algorithm to be applied later and type of the data to be passed into the methodology. With PCA, logistic regression performs better. Since logistic regression is a linear model, the reason may roots in that PCA decreases the relevance between features, which makes features more independent.

For further improvement, we can try different solvers in different models and compare the time cost and the accuracy of the result. Other ensemble learning methods such as XGboost and GBDT may be applied to gain more accurate results.

References:

Rong-En Fan, et al. "Working Set Selection Using Second Order Information for Training Support Vector Machines." *Journal of Machine Learning Research*, vol. 6, no. 63, Crossref Test, Nov. 2005, pp. 1889–918.

Li, Dong-Hui, and Masao Fukushima. "On The Global Convergence of the BFGS Method for Nonconvex Unconstrained Optimization Problems." *SIAM Journal on Optimization*, vol. 11, no. 4, Society for Industrial and Applied Mathematics (SIAM), Jan. 2001, pp. 1054–64. https://doi.org/10.1137/s1052623499354242.

Jonathon Shlens. "A Tutorial on Principal Component Analysis." *arXiv: Learning*, Apr. 2014, cs.gmu.edu/~hrangwal/files/pca.pdf.

Wisconsin Diagnostic Breast Cancer (WDBC) Dataset and Wisconsin Prognostic Breast Cancer (WPBC) Dataset. http://ftp.ics.uci.edu/pub/machine-learning-databases/breast-cancer-wisconsin/

Zhu C, Byrd RH, Lu P, Nocedal J. 1997. Algorithm 778: L-BFGS-B. ACM Transactions on Mathematical Software. 23(4):550–560. doi:10.1145/279232.279236.