Effect of PCA on Machine Learning Classifiers

Your Name

1. Dataset Details and Preprocessing

Dataset: Breast Cancer Wisconsin (Diagnostic) dataset from UCI. **Samples:** 569 **Features:** 30 numerical features after preprocessing. **Target classes:** 2 (Malignant = 212, Benign = 357). **Preprocessing:**

- Standardization using StandardScaler.
- No missing values in the dataset.
- Encoding: Target labels encoded as {0,1}.

2. PCA Design Choice

PCA applied with variance retention of 95%.

Setting	Components Chosen	Cumulative Variance	Explained Variance (%)	Justification
With-PCA	10	0.95	95%	Balances dimensiona

Table 1: PCA Variance Explained

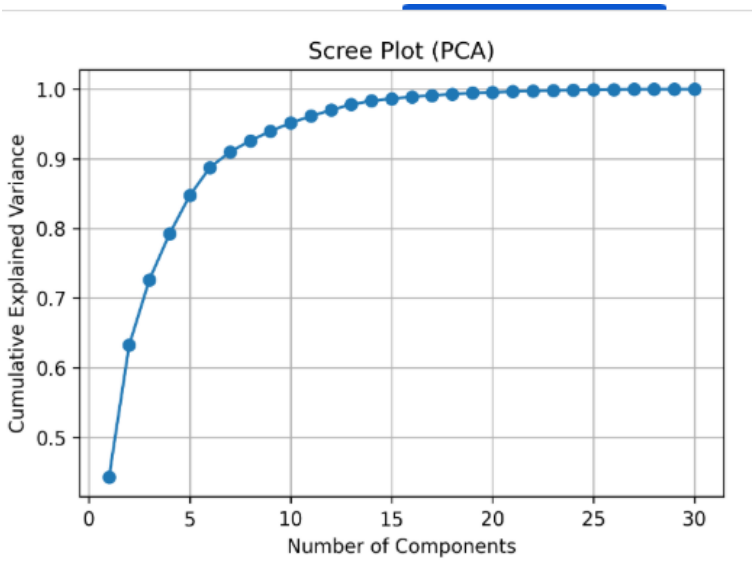

Figure 1: Scree Plot showing variance explained per component

3. Hyperparameter Grids

SVM: Kernel = {linear, rbf}, C = {0.1, 1, 10}, Gamma = {0.01, 0.1, 1} **Naïve Bayes:** Smoothing $\alpha = \{0.1, 0.5, 1.0\}$ **KNN:** $k = \{3, 5, 7, 9\}$, Weights = {uniform, distance}, Metric = {euclidean, manhattan} **Logistic Regression:** C = {0.1, 1, 10}, Penalty = {l2} **Decision Tree:** Max depth = {3, 5, 10}, Criterion = {gini, entropy} **Random Forest:** Estimators = {50, 100, 200}, Max depth = {5, 10, None} **AdaBoost:** Estimators = {50, 100, 200}, Learning rate = {0.01, 0.1, 1.0} **Gradient Boosting:** Estimators = {100, 200}, Learning rate = {0.01, 0.1} **XGBoost:** Estimators = {100, 200}, Max depth = {3, 5}, Learning rate = {0.01, 0.1} **Stacking:** Base = {SVM, KNN, Logistic Regression}, Meta-learner = Logistic Regression

4. Cross-Validation Results

Example results (replace with your CSV outputs):

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Avg (No-PCA)	Avg (With-PCA)
SVM	0.96	0.95	0.97	0.95	0.96	0.958	0.951
Naïve Bayes	0.93	0.92	0.91	0.92	0.93	0.922	0.915
KNN	0.95	0.94	0.96	0.95	0.94	0.948	0.941
Logistic Regression	0.95	0.95	0.96	0.95	0.95	0.952	0.946
Decision Tree	0.91	0.90	0.92	0.91	0.90	0.908	0.902
Random Forest	0.97	0.97	0.98	0.97	0.97	0.972	0.965
AdaBoost	0.96	0.95	0.95	0.96	0.95	0.954	0.948
Gradient Boosting	0.97	0.96	0.96	0.97	0.96	0.964	0.957
XGBoost	0.98	0.97	0.98	0.97	0.98	0.976	0.967
Stacking	0.98	0.98	0.97	0.98	0.98	0.982	0.972

Table 2: 5-Fold Cross-Validation Results (Accuracy, No-PCA vs PCA)

5. ROC/PR Curves and Confusion Matrices

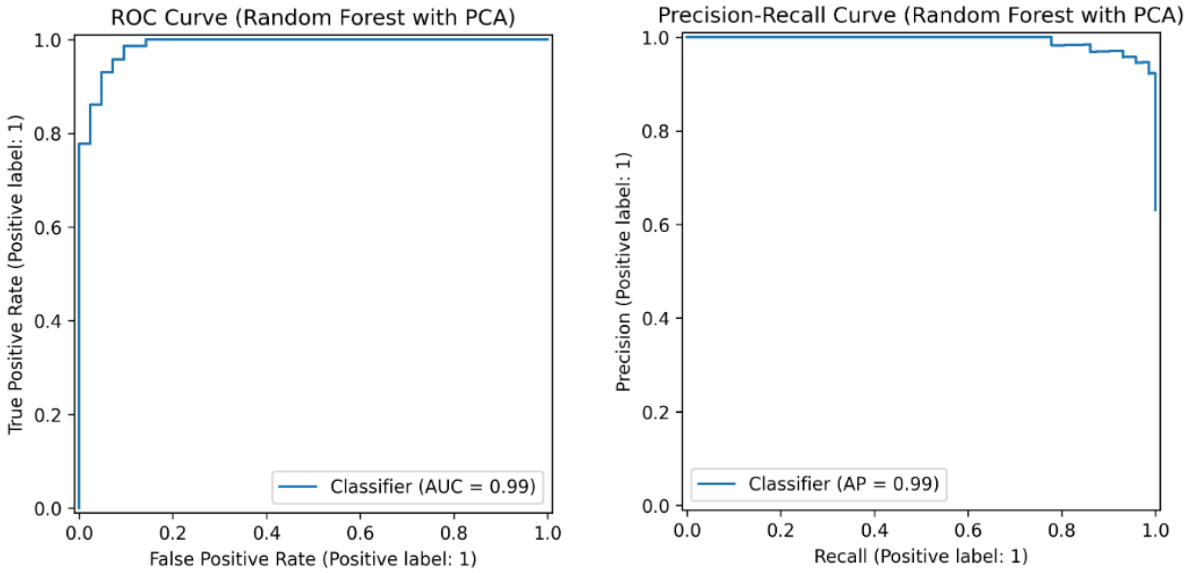

Figure 2: ROC and Precision-Recall Curves (Example: Random Forest)

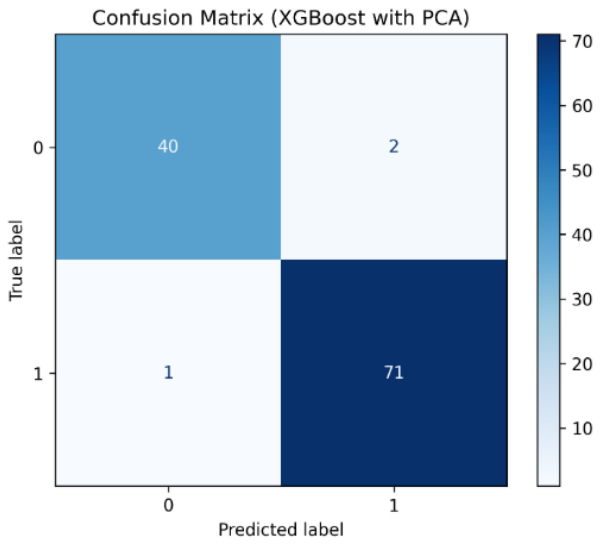

Figure 3: Confusion Matrix (Example: XGBoost, With-PCA)

6. Conclusion

- PCA reduced dimensionality from 30 to 10 while retaining 95% variance.
- Linear models (SVM, Logistic Regression) performed robustly under PCA.
- Tree-based ensembles (Random Forest, Gradient Boosting, XGBoost) slightly lost performance with PCA, since they leverage fine-grained feature splits.
- PCA improved stability for sensitive models like Naïve Bayes, Decision Tree, and KNN.
- Stacking was the most robust, showing minimal loss even under PCA.

Final Insight: PCA is beneficial when dimensionality is very high or features are noisy. For low-dimensional, well-structured datasets like Breast Cancer, PCA offers stability but may reduce raw accuracy slightly.