Dimensionality Reduction Techniques

Final Project

Machine Learning II
Syed Bilal Rizwan

Table of Contents

1. Objective

The aim of this project is to compare different dimensionality reduction techniques and their effect on Machine Learning performance. The techniques are tried on 15 datasets to see the general behavior.

2. Datasets Chosen

2.1 Classification Datasets Description:

1. **Marketing Campaign Dataset**: A response model can provide a significant boost to the efficiency of a marketing campaign by increasing responses or reducing expenses. The objective is to predict who will respond to an offer for a product or service.
2. **Credit Card Fraud Dataset**: The dataset contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.
3. **Heart Disease Prediction Dataset:** According to the CDC, heart disease is one of the leading causes of death for people of most races in the US (African Americans, American Indians and Alaska Natives, and white people). Originally, the dataset come from the CDC and is a major part of the Behavioral Risk Factor Surveillance System (BRFSS), which conducts annual telephone surveys to gather data on the health status of U.S. residents. It consists of 401,958 rows and 279 columns which are reduced to 20 columns.
4. **Diabetes Dataset:** The dataset represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks. It includes over 50 features representing patient and hospital outcomes.
5. **High Income Prediction:** Extraction was done by Barry Becker from the 1994 Census database. A set of clean records was extracted. Prediction task is to determine whether a person makes over 50K a year.
6. **Dry Beans Dataset:** Images of 13,611 grains of 7 different registered dry beans were taken with a high-resolution camera. A total of 16 features; 12 dimensions and 4 shape forms, were obtained from the grains. Prediction task is to find out the type of bean it is.
7. **Banknote Authentication Dataset:** Data were extracted from images that were taken from genuine and forged banknote-like specimens. For digitization, an industrial camera usually used for print inspection was used. The final images have 400x 400 pixels.
8. **Audit Data:** The goal of the research is to help the auditors by building a classification model that can predict the fraudulent firm on the basis the present and historical risk factors.

2.2 Regression Dataset Description

1. **Combined Cycle Power Plant Dataset:** The dataset contains 9568 data points collected from a Combined Cycle Power Plant over 6 years (2006-2011), when the power plant was set to work with full load. Features consist of hourly average ambient variables

Temperature (T), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V) to predict the net hourly electrical energy output (EP) of the plant. Prediction task is to predict the Electrical Energy Output.

2. **Energy Efficiency Dataset:** Perform energy analysis using 12 different building shapes simulated in Ecotect. The buildings differ with respect to the glazing area, the glazing area distribution, and the orientation, amongst other parameters.
3. **QSAR Aquatic Toxicity Dataset:** This dataset was used to develop quantitative regression QSAR models to predict acute aquatic toxicity towards the fish Pimephales promelas (fathead minnow) on a set of 908 chemicals. to predict acute aquatic toxicity towards Daphnia Magna. LC50 data, which is the concentration that causes death in 50% of test D. magna over a test duration of 48 hours, was used as model response.
4. **Bike Sharing Dataset:** Bike sharing systems are new generation of traditional bike rentals where entire process from membership, rental and return has become automatic. Through these systems, user can easily rent a bike from a particular position and return at another position. Currently, there are about over 500 bike-sharing programs around the world which is composed of over 500 thousand bicycles. Today, there exists great interest in these systems due to their key role in traffic, environmental and health issues. Goal is to predict the number of bikes given the other variables.
5. **Wine Quality Dataset:** Goal is to predict the quality of wine given the other variables.
6. **Student Performance Dataset:** This data approach student achievement in secondary education of two Portuguese schools. The data attributes include student grades, demographic, social and school related features) and it was collected by using school reports and questionnaires.
7. **Buzz in social media(Tom's Hardware Dataset):** This dataset contains examples of buzz events from two different social networks: Twitter, and Tom's Hardware, a forum network focusing on modern technology with more conservative dynamics.

3. Background

Training a machine learning model on large datasets require a lot of computational resources and is excessively time consuming as well. To achieve the end-goal in a realistic timeframe, it is important to think of ways to pre-process dataset in a way which leads to less computation and allow scalability. This is where dimensionality reduction techniques come into the picture.

Dimensionality reduction maps a high dimensional dataset into a lower dimensional space without losing information in the dataset. These techniques are used as a pre-processing step before using the dataset for training. There are several renowned DR techniques of which a few are chosen for the comparison analysis below.

3.1 Principal Component Analysis (PCA)

3 different variants of PCA are tried below.

3.1.1 Normal PCA

PCA converts high dimensional dataset into a lower dimensional dataset while still capturing maximum information in the dataset. It does that by converting n correlated features into k uncorrelated features(components) where k is smaller than n. Furthermore, it ensures that maximum variance is captured by the first component and the second highest variance is captured by the second principal component so this way most of the variance is captured by the first few independent components which are then used to transform the dataset into a lower dimension.

3.1.2 Sparse PCA(spca)

Sparse PCA is another variant of PCA that extract sparse components which can help in reconstruction of data. It overcomes the disadvantage of normal PCA which uses all input features to generate the transformed data, but sparse PCA only uses a few input features to transform the data.

3.1.3 Incremental PCA(ipca)

This is like normal PCA however incremental PCA is for large datasets which might be too large to fit to the memory. Incremental PCA makes a low rank approximation which is not based on the number of samples but only on the number of features.

3.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis utilizes class labels along with the dataset to reduce dimensionality making it a supervised dimensionality reduction technique as opposed to PCA. It is a technique that is used to find linear combination of features that ensure separability of classes. Furthermore, the number of components found are always less than number of classes which means it is a strong dimensionality reduction technique. For example, if LDA was applied on a binary classification dataset, then the resulting components would just be 1. Lastly, this technique is only applicable on classification datasets.

3.3 Singular value Decomposition(SVD)

This technique is like PCA where the only difference is that the matrix factorization is performed on data matrix rather than the covariance matrix which is the case for PCA.

4. Methodology

Analysis is done on 15 datasets consisting of 8 classification and 7 regression datasets. The project code structure is divided into these two parts Classification and Regression. Each part is also divided further into more sections. In the first section, the datasets are loaded. In the second section, the datasets are pre-processed and lastly in the last section, each dataset goes through the Machine learning analysis using Dimensionality Reduction Techniques.

4.1 Pre-Processing

Once datasets are loaded, each dataset is pre-processed according to its need. A bird's eye view of original dataset is printed before starting its pre-processing.

1. Firstly, the datatypes and missing values in each column is checked. If any missing values exist, they are addressed by either removing them or imputing them.
2. Unnecessary columns are removed
3. Categorical variables are one-hot encoded
4. All numerical columns are scaled using a Min-Max Scaler
5. If any additional pre-processing is required by any dataset, it is done in the last step.
6. Predictors are separated from target variable. Convention is to name predictors dataframe as dataset_df and target variable as dataset_classes

Finally, the predictors dataset is printed to see how it looks. This process is repeated for all datasets belonging to that section i.e., classification or regression.

4.2 Machine Learning with Dimensionality Reduction

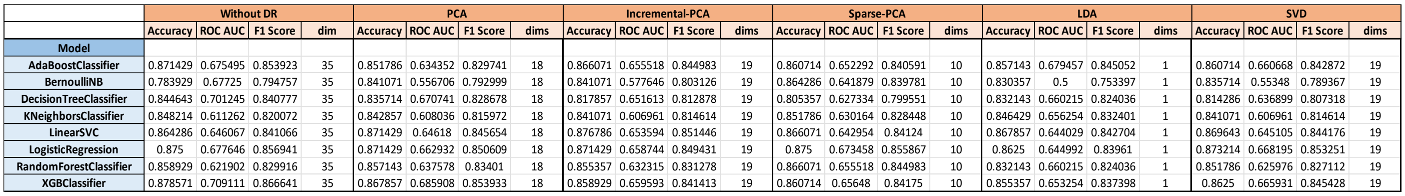
After all datasets belonging to that section are pre-processed, machine learning is carried out by trying out various dimensionality reduction techniques.

1. Firstly, a pipeline is developed to do the whole analysis and give us results dataframe. Then, the pipeline is run on all the datasets of that part. Lazy predict library is used to automate running different machine learning models for the classification task and regression task. The pipeline can be broken into 6 parts:
   i. Lazy Predict on dataset with original features
   ii. Applying PCA and then running Lazy Predict on resulting dataset.
   iii. Applying other PCA variants and then running Lazy Predict on resulting dataset.
   iv. Applying LDA and then running resulting dataset (only applicable for classification datasets)
   v. Applying SVD and then running resulting dataset
   vi. Compiling results from each iteration and output a results dataframe
   vii. Then, each dataset is passed through the pipeline and its results are exported into an excel sheet.
2. Results are printed on the notebook and a detailed analysis is done for the results of that dataset.

5. Results and Analysis

5.1 Classification Datasets

5.1.1 Marketing Dataset



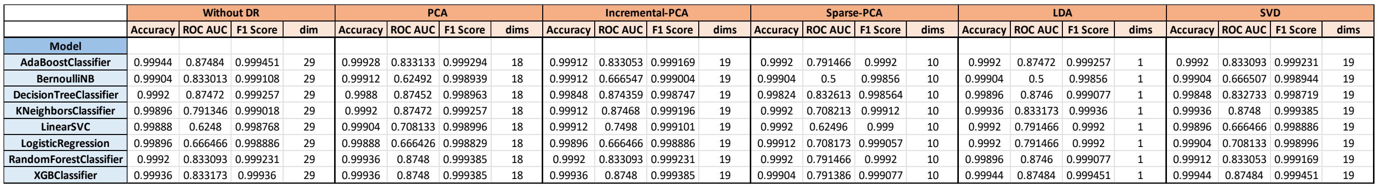
	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
AdaBoostClassifier	0.871429	0.675495	0.853923	35	0.851786	0.634352	0.829741	18	0.866071	0.655518	0.844983	19	0.860714	0.652292	0.840591	10	0.857143	0.679457	0.845052	1	0.860714	0.660668	0.842872	19
BernoulliNB	0.783929	0.67725	0.794757	35	0.841071	0.556706	0.792999	18	0.841071	0.577646	0.803126	19	0.864286	0.641879	0.839781	10	0.830357	0.5	0.753397	1	0.835714	0.55348	0.789367	19
DecisionTreeClassifier	0.844643	0.701245	0.840777	35	0.835714	0.670741	0.828678	18	0.817857	0.651613	0.812878	19	0.805357	0.627334	0.799551	10	0.832143	0.660215	0.824036	1	0.814286	0.636899	0.807318	19
KNeighborsClassifier	0.848214	0.611262	0.820072	35	0.842857	0.608036	0.815972	18	0.841071	0.606961	0.814614	19	0.851786	0.630164	0.828448	10	0.846429	0.656254	0.832401	1	0.841071	0.606961	0.814614	19
LinearSVC	0.864286	0.646067	0.841066	35	0.871429	0.64618	0.845654	18	0.876786	0.653594	0.851446	19	0.866071	0.642954	0.84124	10	0.867857	0.644029	0.842704	1	0.869643	0.645105	0.844176	19
LogisticRegression	0.875	0.677646	0.856941	35	0.871429	0.662932	0.850609	18	0.871429	0.658744	0.849431	19	0.875	0.673458	0.855867	10	0.8625	0.644992	0.83961	1	0.873214	0.668195	0.853251	19
RandomForestClassifier	0.858929	0.621902	0.829916	35	0.857143	0.637578	0.83401	18	0.855357	0.632315	0.831278	19	0.866071	0.655518	0.844983	10	0.832143	0.660215	0.824036	1	0.851786	0.625976	0.827112	19
XGBClassifier	0.878571	0.709111	0.866641	35	0.867857	0.685908	0.853933	18	0.858929	0.659593	0.841413	19	0.860714	0.65648	0.84175	10	0.855357	0.653254	0.837398	1	0.8625	0.665931	0.845428	19

Analysis:

1. For the marketing dataset, initially with full dataset having 35 features gave us the maximum accuracy of 0.88 and F1 score of 0.87 from the XG Boost Algorithm. For Naive bayes, we saw an accuracy of 0.78 and F1 score of 0.79.
2. After applying normal PCA, the accuracy and F1 score are the same as without applying PCA which means that all variation was captured by the Principal Components. Furthermore, Naive Bayes algorithm performance increased greatly when feature reduction was applied as its performance increased from 0.76 to 0.84. However, it is important to note that AUC-ROC of the algorithm dropped.
3. Applying the other PCA variants such as Incremental and Sparse PCA does not improve the results and they perform at Par with PCA although sparse PCA reduced the dimensions further to 10 with truly little sacrifice to variance capture. Naive Bayes algorithm's performance increased further as well. However, it is important to note that AUC-ROC of the algorithm dropped.
4. Then Linear Discriminant Analysis was tried, and the performance was at Par with the other DR techniques. However, since it reduces the dimensions to 1, it proves to be the best DR technique so far.
5. Lastly, Singular Value Decomposition also performed at PAR with the other DR techniques and gave promising results.

From this dataset's perspective LDA performs well since it reduces the dimensions to just 1 with truly little compromise on variance capture.

5.1.2 Credit Card Fraud Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
AdaBoostClassifier	0.99944	0.87484	0.999451	29	0.99928	0.833133	0.999294	18	0.99912	0.833053	0.999169	19	0.9992	0.791466	0.9992	10	0.9992	0.87472	0.999257	1	0.9992	0.833093	0.999231	19
BernoulliNB	0.99904	0.833013	0.999108	29	0.99912	0.62492	0.998939	18	0.99912	0.666547	0.999004	19	0.99904	0.5	0.99856	10	0.99904	0.5	0.99856	1	0.99904	0.666507	0.998944	19
DecisionTreeClassifier	0.9992	0.87472	0.999257	29	0.9988	0.87452	0.998963	18	0.99848	0.874359	0.998747	19	0.99824	0.832613	0.998564	10	0.99896	0.8746	0.999077	1	0.99848	0.832733	0.998719	19
KNeighborsClassifier	0.99896	0.791346	0.999018	29	0.9992	0.87472	0.999257	18	0.99912	0.87468	0.999196	19	0.9992	0.708213	0.99912	10	0.99936	0.833173	0.99936	1	0.99936	0.8748	0.999385	19
LinearSVC	0.99888	0.6248	0.998768	29	0.99904	0.708133	0.998996	18	0.99912	0.7498	0.999101	19	0.9992	0.62496	0.999	10	0.9992	0.791466	0.9992	1	0.99896	0.666466	0.998886	19
LogisticRegression	0.99896	0.666466	0.998886	29	0.99888	0.666426	0.998829	18	0.99896	0.666466	0.998886	19	0.99912	0.708173	0.999057	10	0.9992	0.791466	0.9992	1	0.99904	0.708133	0.998996	19
RandomForestClassifier	0.9992	0.833093	0.999231	29	0.99936	0.8748	0.999385	18	0.9992	0.833093	0.999231	19	0.9992	0.791466	0.9992	10	0.99896	0.8746	0.999077	1	0.99912	0.833053	0.999169	19
XGBClassifier	0.99936	0.833173	0.99936	29	0.99936	0.8748	0.999385	18	0.99936	0.8748	0.999385	19	0.99904	0.791386	0.999077	10	0.99944	0.87484	0.999451	1	0.99944	0.87484	0.999451	19

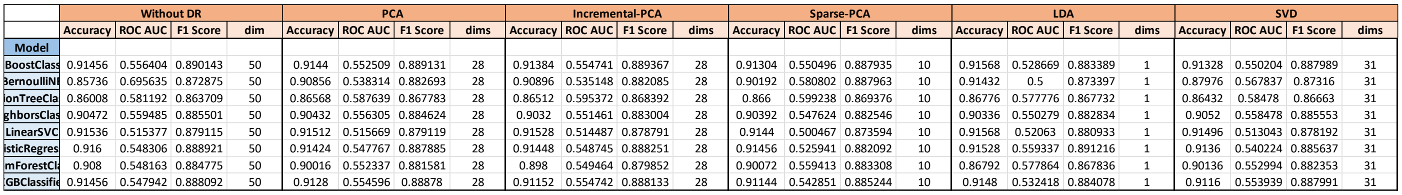
Analysis:

DR Techniques were tried on Credit Card dataset and the results are summarized below:

1. Accuracy and F1-measure of this dataset cannot be compared because it is remarkably close to 1. This is because the dataset is highly imbalanced with 0.15% of fraud classes. Therefore AUC-ROC is a better metric to gauge for this dataset.
2. The best AUC-ROC achieved without any DR technique was 0.87 which is kept as a benchmark to compare.
3. When normal PCA is applied, some algorithm's AUC-ROC increased such as Support Vector Machine and Random Forest. The best AUC-ROC remained the same at 0.87 which showed PCA to be a good technique capturing all the variance.
4. Then, when more PCA variants were tried, the performance was at par for incremental PCA but for Sparse PCA, the AUC-ROC dropped which shows that reducing the dimensions too much to 10 compromises on variance capture.
5. However, for LDA on this dataset, the performance was at par with full dataset which makes it the best DR technique as far as it reduced the dimension to 1.
6. For SVD, some algorithms suffered dips, but most algorithms performed at par with the without DR which proves it is a good technique as well.

It is evident that LDA has performed good for this dataset as it captures all variance while reducing the total dimensions to 1.

5.1.3 Heart Disease Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
BoostClass	0.91456	0.556404	0.890143	50	0.9144	0.552509	0.889131	28	0.91384	0.554741	0.889367	28	0.91304	0.550496	0.887935	10	0.91568	0.528669	0.883389	1	0.91328	0.550204	0.887989	31
BernoulliNI	0.85736	0.695635	0.872875	50	0.90856	0.538314	0.882693	28	0.90896	0.535148	0.882085	28	0.90192	0.580802	0.887963	10	0.91432	0.5	0.873397	1	0.87976	0.567837	0.87316	31
ionTreeCla	0.86008	0.581192	0.863709	50	0.86568	0.587639	0.867783	28	0.86512	0.595372	0.868392	28	0.866	0.599238	0.869376	10	0.86776	0.577776	0.867732	1	0.86432	0.58478	0.86663	31
ghborsClas	0.90472	0.559485	0.885501	50	0.90432	0.556305	0.884624	28	0.9032	0.551461	0.883004	28	0.90392	0.547624	0.882546	10	0.90336	0.550279	0.882834	1	0.9052	0.558478	0.885553	31
LinearSVC	0.91536	0.515377	0.879115	50	0.91512	0.515669	0.879119	28	0.91528	0.514487	0.878791	28	0.9144	0.500467	0.873594	10	0.91568	0.52063	0.880933	1	0.91496	0.513043	0.878192	31
isticRegres	0.916	0.548306	0.888921	50	0.91424	0.547767	0.887885	28	0.91448	0.548745	0.888251	28	0.91456	0.525941	0.882092	10	0.91528	0.559337	0.891216	1	0.9136	0.540224	0.885637	31
mForestCla	0.908	0.548163	0.884775	50	0.90016	0.552337	0.881581	28	0.898	0.549464	0.879852	28	0.90072	0.559413	0.883308	10	0.86792	0.577864	0.867836	1	0.90136	0.552994	0.882353	31
GBClassifie	0.91456	0.547942	0.888092	50	0.9128	0.554596	0.88878	28	0.91152	0.554742	0.888133	28	0.91144	0.542851	0.885244	10	0.9148	0.532418	0.884078	1	0.9116	0.553939	0.887991	31

Analysis:

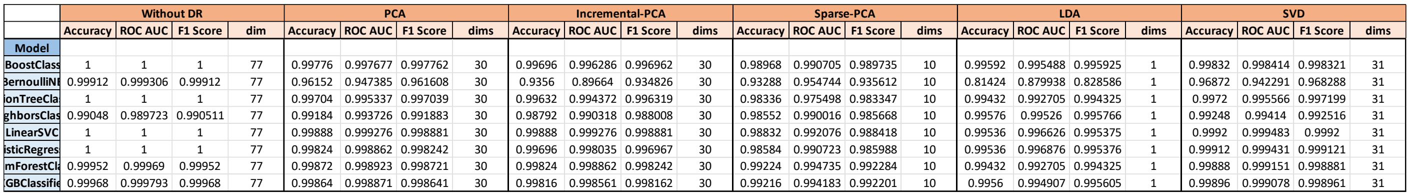
Since the heart disease dataset is slightly imbalanced, we can use AUC-ROC or F1 score to gauge its performance. I would be making comparison based on both.

1. Without DR, the best F1-score is achieved to be 0.89 by several algorithms with an AUC-ROC of 0.56. This would be used as a benchmark for further comparisons.
2. When PCA is applied, the F1-score and AUC-ROC were still the same as without DR which proved to be a good DR technique as it reduces the dims from 50 to 28.

3. Similarly, for other PCA variants, they all performed at PAR however, sparse PCA performs better in a way that it does not compromise on variance capture and reduces dimensions the most which is 10.
4. When LDA is applied, there is barely any compromise on variance capture as the metrics remain same however, the dimensions decrease to 1 which is its major achievement.
5. SVD performs at par with the other techniques however, the dims are still too high compared to the others.

It is becoming evident that LDA is the winner DR technique!

5.1.4 Diabetes Dataset

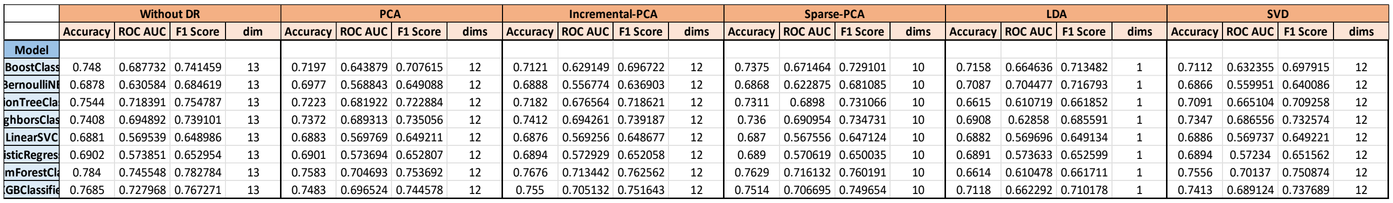


	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
BoostClass	1	1	1	77	0.99776	0.997677	0.997762	30	0.99696	0.996286	0.996962	30	0.98968	0.990705	0.989735	10	0.99592	0.995488	0.995925	1	0.99832	0.998414	0.998321	31
BernoulliNI	0.99912	0.999306	0.99912	77	0.96152	0.947385	0.961608	30	0.9356	0.89664	0.934826	30	0.93288	0.954744	0.935612	10	0.81424	0.879938	0.828586	1	0.96872	0.942291	0.968288	31
ionTreeCla	1	1	1	77	0.99704	0.995337	0.997039	30	0.99632	0.994372	0.996319	30	0.98336	0.975498	0.983347	10	0.99432	0.992705	0.994325	1	0.9972	0.995566	0.997199	31
ghborsClas	0.99048	0.989723	0.990511	77	0.99184	0.993726	0.991883	30	0.98792	0.990318	0.988008	30	0.98552	0.990016	0.985668	10	0.99576	0.99526	0.995766	1	0.99248	0.99414	0.992516	31
LinearSVC	1	1	1	77	0.99888	0.999276	0.998881	30	0.99888	0.999276	0.998881	30	0.98832	0.992076	0.988418	10	0.99536	0.996626	0.995375	1	0.9992	0.999483	0.9992	31
isticRegres	1	1	1	77	0.99824	0.998862	0.998242	30	0.99696	0.998035	0.996967	30	0.98584	0.990723	0.985988	10	0.99536	0.996876	0.995376	1	0.99912	0.999431	0.999121	31
mForestCla	0.99952	0.99969	0.99952	77	0.99872	0.998923	0.998721	30	0.99824	0.998862	0.998242	30	0.99224	0.994735	0.992284	10	0.99432	0.992705	0.994325	1	0.99888	0.999151	0.998881	31
GBClassifie	0.99968	0.999793	0.99968	77	0.99864	0.998871	0.998641	30	0.99816	0.998561	0.998162	30	0.99216	0.994183	0.992201	10	0.9956	0.994907	0.995605	1	0.99896	0.999078	0.998961	31

Analysis:

There cannot be much analysis done on this dataset because all its values are 1 or close to 1 which is too good to be true. It is possible that the sub sample of dataset taken may have very few positive labels which would make it a highly imbalanced dataset and more prone to predicting the same class. But, in any case, it can be noticed that all the DR Techniques perform the same as non-DR one and LDA reduces the dimensions the most, so it still is the winner DR technique.

5.1.5 High Income Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
BoostClass	0.748	0.687732	0.741459	13	0.7197	0.643879	0.707615	12	0.7121	0.629149	0.696722	12	0.7375	0.671464	0.729101	10	0.7158	0.664636	0.713482	1	0.7112	0.632355	0.697915	12
BernoulliNI	0.6878	0.630584	0.684619	13	0.6977	0.568843	0.649088	12	0.6888	0.556774	0.636903	12	0.6868	0.622875	0.681085	10	0.7087	0.704477	0.716793	1	0.6866	0.559951	0.640086	12
ionTreeCla	0.7544	0.718391	0.754787	13	0.7223	0.681922	0.722884	12	0.7182	0.676564	0.718621	12	0.7311	0.6898	0.731066	10	0.6615	0.610719	0.661852	1	0.7091	0.665104	0.709258	12
ghborsClas	0.7408	0.694892	0.739101	13	0.7372	0.689313	0.735056	12	0.7412	0.694261	0.739187	12	0.736	0.690954	0.734731	10	0.6908	0.62858	0.685591	1	0.7347	0.686556	0.732574	12
LinearSVC	0.6881	0.569539	0.648986	13	0.6883	0.569769	0.649211	12	0.6876	0.569256	0.648677	12	0.687	0.567556	0.647124	10	0.6882	0.569696	0.649134	1	0.6886	0.569737	0.649221	12
isticRegres	0.6902	0.573851	0.652954	13	0.6901	0.573694	0.652807	12	0.6894	0.572929	0.652058	12	0.689	0.570619	0.650035	10	0.6891	0.573633	0.652599	1	0.6894	0.57234	0.651562	12
mForestCla	0.784	0.745548	0.782784	13	0.7583	0.704693	0.753692	12	0.7676	0.713442	0.762562	12	0.7629	0.716132	0.760191	10	0.6614	0.610478	0.661711	1	0.7556	0.70137	0.750874	12
GBClassifie	0.7685	0.727968	0.767271	13	0.7483	0.696524	0.744578	12	0.755	0.705132	0.751643	12	0.7514	0.706695	0.749654	10	0.7118	0.662292	0.710178	1	0.7413	0.689124	0.737689	12

Analysis:

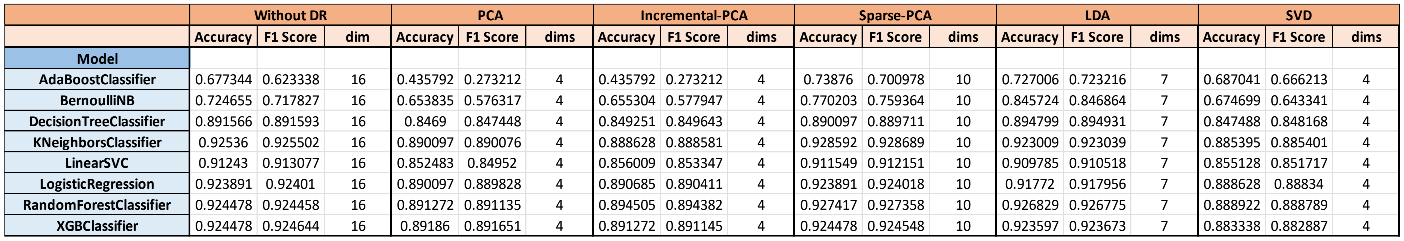
The dataset is not that imbalanced so we can use any of the three metrics to compare the DR techniques. To keep it coherent, lets focus on F1 score for this dataset:

1. The best F1 score achieved with full dataset without any DR technique is around 0.78 given by Random Forest.
2. After applying PCA, a very slight drop in F1 score is seen however, the dims are not reduced that much either and remain at 12 reduced from 13. So, at the compromise of little variance capture, only 1 feature is reduced.

3. Applying different variants of PCA resulted in comparable results. However, sparse PCA performs better in a way that it reduces dimensions from 12 to 10 with truly little compromise on variance capture.
4. After applying LDA, the features are reduced to 1 however, there is a significant drop in variance capture which shows that LDA does not perform good on this dataset.
5. Lastly, SVD performed like PCA.

It is seen that in this dataset, DR techniques cause a significant decrease in variance capture if features are reduced to less and the conclusion reached from this is that the dataset already contains the most important information.

5.1.6 Dry Beans Dataset



	Without DR			PCA			Incremental-PCA			Sparse-PCA			LDA			SVD		
	Accuracy	F1 Score	dim	Accuracy	F1 Score	dims	Accuracy	F1 Score	dims	Accuracy	F1 Score	dims	Accuracy	F1 Score	dims	Accuracy	F1 Score	dims
Model																		
AdaBoostClassifier	0.677344	0.623338	16	0.435792	0.273212	4	0.435792	0.273212	4	0.73876	0.700978	10	0.727006	0.723216	7	0.687041	0.666213	4
BernoulliNB	0.724655	0.717827	16	0.653835	0.576317	4	0.655304	0.577947	4	0.770203	0.759364	10	0.845724	0.846864	7	0.674699	0.643341	4
DecisionTreeClassifier	0.891566	0.891593	16	0.8469	0.847448	4	0.849251	0.849643	4	0.890097	0.889711	10	0.894799	0.894931	7	0.847488	0.848168	4
KNeighborsClassifier	0.92536	0.925502	16	0.890097	0.890076	4	0.888628	0.888581	4	0.928592	0.928689	10	0.923009	0.923039	7	0.885395	0.885401	4
LinearSVC	0.91243	0.913077	16	0.852483	0.84952	4	0.856009	0.853347	4	0.911549	0.912151	10	0.909785	0.910518	7	0.855128	0.851717	4
LogisticRegression	0.923891	0.92401	16	0.890097	0.889828	4	0.890685	0.890411	4	0.923891	0.924018	10	0.91772	0.917956	7	0.888628	0.88834	4
RandomForestClassifier	0.924478	0.924458	16	0.891272	0.891135	4	0.894505	0.894382	4	0.927417	0.927358	10	0.926829	0.926775	7	0.888922	0.888789	4
XGBClassifier	0.924478	0.924644	16	0.89186	0.891651	4	0.891272	0.891145	4	0.924478	0.924548	10	0.923597	0.923673	7	0.883338	0.882887	4

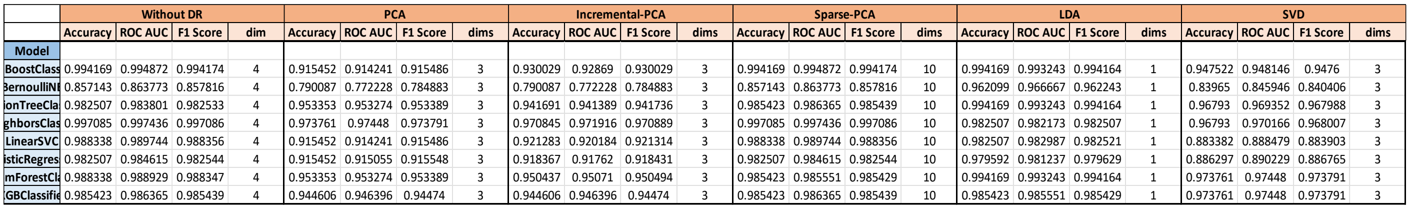
Analysis:

For the dry bean's dataset, AUC-ROC was removed since it is a multi-class problem. Since the dataset is balanced, I will be focusing on Accuracy as the metric for comparison.

1. The best accuracy achieved for this dataset without any DR technique is 0.92.
2. After applying PCA, the accuracy slightly decreased but the dimensions went from 16 to 4 which is good.
3. After applying variants of PCA, the accuracy remained at Par with normal PCA. However, sparse PCA gave a slightly better accuracy of 0.93 and reduced features to 10 from 16.
4. After applying LDA, the dimensions were reduced to 7 which is exceptionally good with no expense to accuracy and variance capture. LDA proves to be good so far.
5. SVD reduced dimensions to 4 but there was a slight drop in accuracy.

LDA still proves to be good as it maintains the best accuracy with significant reduction of features. The other DR techniques are still good enough.

5.1.7 Banknote Authentication Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
BoostClass	0.994169	0.994872	0.994174	4	0.915452	0.914241	0.915486	3	0.930029	0.92869	0.930029	3	0.994169	0.994872	0.994174	10	0.994169	0.993243	0.994164	1	0.947522	0.948146	0.9476	3
BernoulliNI	0.857143	0.863773	0.857816	4	0.790087	0.772228	0.784883	3	0.790087	0.772228	0.784883	3	0.857143	0.863773	0.857816	10	0.962099	0.966667	0.962243	1	0.83965	0.845946	0.840406	3
ionTreeCla	0.982507	0.983801	0.982533	4	0.953353	0.953274	0.953389	3	0.941691	0.941389	0.941736	3	0.985423	0.986365	0.985439	10	0.994169	0.993243	0.994164	1	0.96793	0.969352	0.967988	3
ghborsClas	0.997085	0.997436	0.997086	4	0.973761	0.97448	0.973791	3	0.970845	0.971916	0.970889	3	0.997085	0.997436	0.997086	10	0.982507	0.982173	0.982507	1	0.96793	0.970166	0.968007	3
LinearSVC	0.988338	0.989744	0.988356	4	0.915452	0.914241	0.915486	3	0.921283	0.920184	0.921314	3	0.988338	0.989744	0.988356	10	0.982507	0.982987	0.982521	1	0.883382	0.888479	0.883903	3
isticRegres	0.982507	0.984615	0.982544	4	0.915452	0.915055	0.915548	3	0.918367	0.91762	0.918431	3	0.982507	0.984615	0.982544	10	0.979592	0.981237	0.979629	1	0.886297	0.890229	0.886765	3
mForestCla	0.988338	0.988929	0.988347	4	0.953353	0.953274	0.953389	3	0.950437	0.95071	0.950494	3	0.985423	0.985551	0.985429	10	0.994169	0.993243	0.994164	1	0.973761	0.97448	0.973791	3
GBClassifie	0.985423	0.986365	0.985439	4	0.944606	0.946396	0.94474	3	0.944606	0.946396	0.94474	3	0.985423	0.986365	0.985439	10	0.985423	0.985551	0.985429	1	0.973761	0.97448	0.973791	3

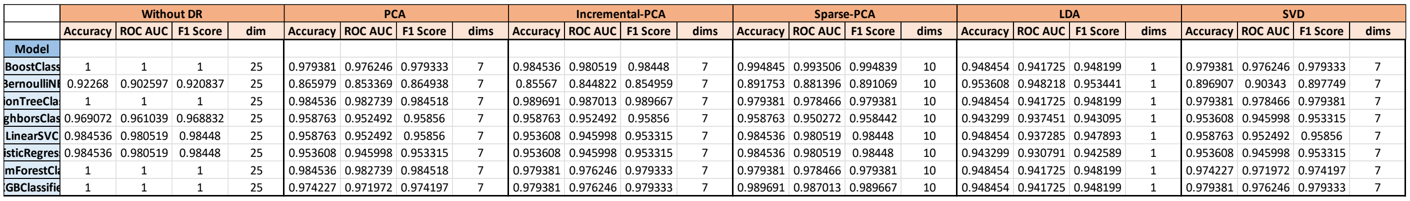
Analysis:

This dataset is a banknote authentication dataset. Since the dataset is balanced, we can use accuracy as a metric to gauge it:

1. The accuracy achieved without DR techniques is 1 which seems that the model is too good to be true.
2. After applying PCA, the accuracy is slightly compromised to 0.97 with just 1 feature reduction.
3. Different PCA variants perform similar with sparse PCA performing worse as it increases the dimensions from 4 to 10.
4. LDA captures all the variance of the dataset and does not compromise on accuracy while reducing the number of dimensions from 4 to 1 which shows that it performs very well.
5. SVD performs at par with PCA.

It can be seen here as well that LDA performs well for this dataset as it reduces number of dimensions to 1 without any compromise in accuracy.

5.1.8 Audit Risk Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				LDA				SVD			
	Accuracy	ROC AUC	F1 Score	dim	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims	Accuracy	ROC AUC	F1 Score	dims
Model																								
BoostClass	1	1	1	25	0.979381	0.976246	0.979333	7	0.984536	0.980519	0.98448	7	0.994845	0.993506	0.994839	10	0.948454	0.941725	0.948199	1	0.979381	0.976246	0.979333	7
BernoulliNI	0.92268	0.902597	0.920837	25	0.865979	0.853369	0.864938	7	0.85567	0.844822	0.854959	7	0.891753	0.881396	0.891069	10	0.953608	0.948218	0.953441	1	0.896907	0.90343	0.897749	7
ionTreeCla	1	1	1	25	0.984536	0.982739	0.984518	7	0.989691	0.987013	0.989667	7	0.979381	0.978466	0.979381	10	0.948454	0.941725	0.948199	1	0.979381	0.978466	0.979381	7
ghborsClas	0.969072	0.961039	0.968832	25	0.958763	0.952492	0.95856	7	0.958763	0.952492	0.95856	7	0.958763	0.950272	0.958442	10	0.943299	0.937451	0.943095	1	0.953608	0.945998	0.953315	7
LinearSVC	0.984536	0.980519	0.98448	25	0.958763	0.952492	0.95856	7	0.953608	0.945998	0.953315	7	0.984536	0.980519	0.98448	10	0.948454	0.937285	0.947893	1	0.958763	0.952492	0.95856	7
isticRegres	0.984536	0.980519	0.98448	25	0.953608	0.945998	0.953315	7	0.953608	0.945998	0.953315	7	0.984536	0.980519	0.98448	10	0.943299	0.930791	0.942589	1	0.953608	0.945998	0.953315	7
mForestCla	1	1	1	25	0.984536	0.982739	0.984518	7	0.979381	0.976246	0.979333	7	0.979381	0.978466	0.979381	10	0.948454	0.941725	0.948199	1	0.974227	0.971972	0.974197	7
GBClassifie	1	1	1	25	0.974227	0.971972	0.974197	7	0.979381	0.976246	0.979333	7	0.989691	0.987013	0.989667	10	0.948454	0.941725	0.948199	1	0.979381	0.976246	0.979333	7

Analysis:

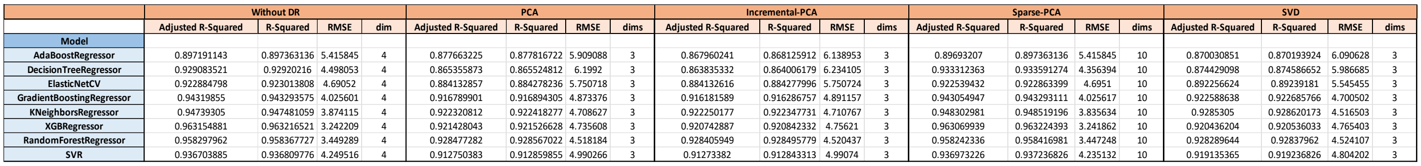
The audit risk dataset is a simple dataset therefore it has particularly good metrics. Since the dataset is balanced, I would focus on looking at accuracy for comparison:

1. The best accuracy achieved is 1.0 without applying any DR techniques. The total dims are 25.
2. After applying PCA, the accuracy slightly decreased to around 0.98 while reducing features significantly to 7 dimensions.
3. After trying other PCA variants, the results were at par with PCA with accuracy around 0.99.
4. After applying LDA, the best accuracy achieved was 0.95 which is slightly less than 1 however the number of dimensions become 1 which is a significant achievement.
5. SVD gives an accuracy of 0.98 and reduces dimensions to 7 which is the same as PCA.

All techniques perform quite well on this dataset. LDA compromises slightly on variance capture however it reduces dimensionality the most.

5.2 Regression Datasets

5.2.1 Cycle Power Plant Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.897191143	0.897363136	5.415845	4	0.877663225	0.877816722	5.909088	3	0.867960241	0.868125912	6.138953	3	0.89693207	0.897363136	5.415845	10	0.870030851	0.870193924	6.090628	3
DecisionTreeRegressor	0.929083521	0.92920216	4.498053	4	0.865355873	0.865524812	6.1992	3	0.863835332	0.864006179	6.234105	3	0.933312363	0.933591274	4.356394	10	0.874429098	0.874586652	5.986685	3
ElasticNetCV	0.922884798	0.923013808	4.69052	4	0.884132857	0.884278236	5.750718	3	0.884132616	0.884277996	5.750724	3	0.922539432	0.922863399	4.6951	10	0.892256624	0.89239181	5.545455	3
GradientBoostingRegressor	0.94319855	0.943293575	4.025601	4	0.916789901	0.916894305	4.873376	3	0.916181589	0.916286757	4.891157	3	0.943054947	0.943293111	4.025617	10	0.922588638	0.922685766	4.700502	3
KNeighborsRegressor	0.94739305	0.947481059	3.874115	4	0.922320812	0.922418277	4.708627	3	0.922250177	0.922347731	4.710767	3	0.948302981	0.948519196	3.835634	10	0.9285305	0.928620173	4.516503	3
XGBRegressor	0.963154881	0.963216521	3.242209	4	0.921428043	0.921526628	4.735608	3	0.920742887	0.920842332	4.75621	3	0.963069939	0.963224393	3.241862	10	0.920436204	0.920536033	4.765403	3
RandomForestRegressor	0.958297962	0.958367727	3.449289	4	0.928477282	0.928567022	4.518184	3	0.928405949	0.928495779	4.520437	3	0.958242336	0.958416981	3.447248	10	0.928289644	0.92837962	4.524107	3
SVR	0.936703885	0.936809776	4.249516	4	0.912750383	0.912859855	4.990266	3	0.91273382	0.912843313	4.99074	3	0.936973226	0.937236826	4.235132	10	0.919135365	0.919236826	4.804202	3

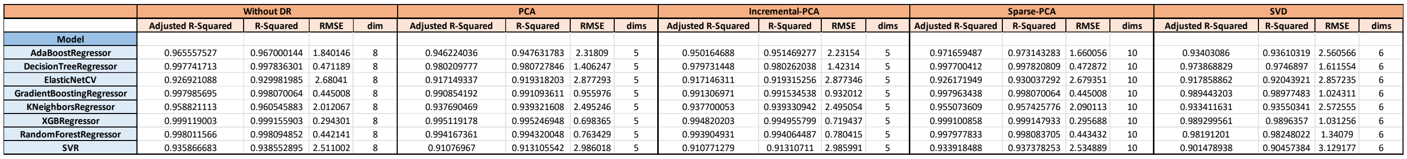
Analysis:

For our analysis, I will be using adjusted R2 to make comparison since number of features are changing for each model:

1. When no DR technique is applied, the best adjusted R2 score achieved was 0.96.
2. After applying PCA, the R2 score slightly decreased to 0.93 with a reduction in features to 3 from 4.
3. When applying other PCA variants, Incremental PCA behaved like PCA, but sparse PCA captured all the variance however it increased the dimensionality rather than decreasing it.
4. When SVD was tried, the variance captured was 0.93 slightly less than 0.96 with 1 feature reduction so it performed at par with PCA.

Only 1 feature was reduced in these DR techniques with little compromise to variance capture, PCA and SVD performed at par with each other.

5.2.2 Energy Efficiency Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.965557527	0.967000144	1.840146	8	0.946224036	0.947631783	2.31809	5	0.950164688	0.951469277	2.23154	5	0.971659487	0.973143283	1.660056	10	0.93403086	0.93610319	2.560566	6
DecisionTreeRegressor	0.997741713	0.997836301	0.471189	8	0.980209777	0.980727846	1.406247	5	0.979731448	0.980262038	1.42314	5	0.997700412	0.997820809	0.472872	10	0.973868829	0.9746897	1.611554	6
ElasticNetCV	0.926921088	0.929981985	2.68041	8	0.917149337	0.919318203	2.877293	5	0.917146311	0.919315256	2.877346	5	0.926171949	0.930037292	2.679351	10	0.917858862	0.92043921	2.857235	6
GradientBoostingRegressor	0.997985695	0.998070064	0.445008	8	0.990854192	0.991093611	0.955976	5	0.991306971	0.991534538	0.932012	5	0.997963438	0.998070064	0.445008	10	0.989443203	0.98977483	1.024311	6
KNeighborsRegressor	0.958821113	0.960545883	2.012067	8	0.937690469	0.939321608	2.495246	5	0.937700053	0.939330942	2.495054	5	0.955073609	0.957425776	2.090113	10	0.933411631	0.93550341	2.572555	6
XGBRegressor	0.999119003	0.999155903	0.294301	8	0.995119178	0.995246948	0.698365	5	0.994820203	0.994955799	0.719437	5	0.999100858	0.999147933	0.295688	10	0.989299561	0.9896357	1.031256	6
RandomForestRegressor	0.998011566	0.998094852	0.442141	8	0.994167361	0.994320048	0.763429	5	0.993904931	0.994064487	0.780415	5	0.997977833	0.998083705	0.443432	10	0.98191201	0.98248022	1.34079	6
SVR	0.935866683	0.938552895	2.511002	8	0.91076967	0.913105542	2.986018	5	0.910771279	0.91310711	2.985991	5	0.933918488	0.937378253	2.534889	10	0.901478938	0.90457384	3.129177	6

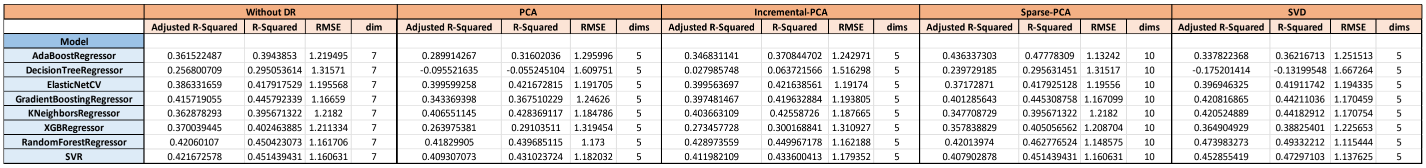
Analysis:

Adjusted R2 score is used to compare the results for the energy efficiency dataset:

1. When no DR technique was applied, the best Adjusted R2 score achieved was 1 which means the model captures all the variance available.
2. When PCA is applied, the features reduced to 5 from 8 but the best adjusted R2 score remained at 1 which shows PCA to be a good technique for this dataset.
3. As other PCA variants are tried, incremental PCA performs at par with PCA, but Sparse PCA increases dimension, so it performs poorly.
4. SVD reduced the dimensions dataset dimensions to 6 from 8 only with almost no compromise on variance capture.

PCA seems to perform well for this dataset as it reduces the maximum dimensions and maintains full variance.

5.2.3 QSAR Aquatic Toxicity Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.361522487	0.3943853	1.219495	7	0.289914267	0.31602036	1.295996	5	0.346831141	0.370844702	1.242971	5	0.436337303	0.47778309	1.13242	10	0.337822368	0.36216713	1.251513	5
DecisionTreeRegressor	0.256800709	0.295053614	1.31571	7	-0.095521635	-0.055245104	1.609751	5	0.027985748	0.063721566	1.516298	5	0.239729185	0.295631451	1.31517	10	-0.175201414	-0.13199548	1.667264	5
ElasticNetCV	0.386331659	0.417917529	1.195568	7	0.399599258	0.421672815	1.191705	5	0.399563697	0.421638561	1.19174	5	0.37172871	0.417925128	1.19556	10	0.396946325	0.41911742	1.194335	5
GradientBoostingRegressor	0.415719055	0.445792339	1.16659	7	0.343369398	0.367510229	1.24626	5	0.397481467	0.419632884	1.193805	5	0.401285643	0.445308758	1.167099	10	0.420816865	0.44211036	1.170459	5
KNeighborsRegressor	0.362878293	0.395671322	1.2182	7	0.406551145	0.428369117	1.184786	5	0.403663109	0.42558726	1.187665	5	0.347708729	0.395671322	1.2182	10	0.420524889	0.44182912	1.170754	5
XGBRegressor	0.370039445	0.402463885	1.211334	7	0.263975381	0.29103511	1.319454	5	0.273457728	0.300168841	1.310927	5	0.357838829	0.405056562	1.208704	10	0.364904929	0.38825401	1.225653	5
RandomForestRegressor	0.42060107	0.450423073	1.161706	7	0.41829905	0.439685115	1.173	5	0.428973559	0.449967178	1.162188	5	0.42013974	0.462776524	1.148575	10	0.473983273	0.49332212	1.115444	5
SVR	0.421672578	0.451439431	1.160631	7	0.409307073	0.431023724	1.182032	5	0.411982109	0.433600413	1.179352	5	0.407902878	0.451439431	1.160631	10	0.452855419	0.47297103	1.137625	5

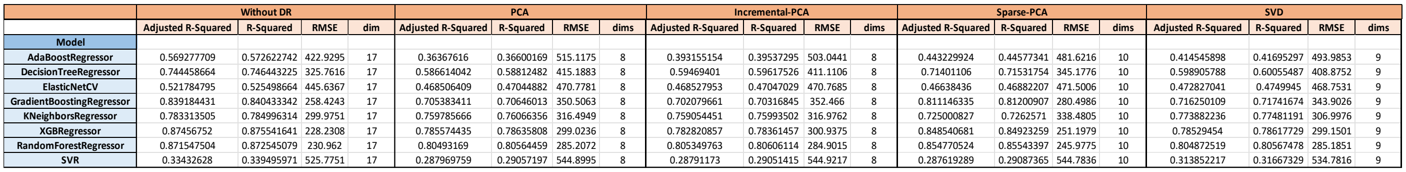
Analysis:

For the aquatic toxicity dataset, adjusted R2 score is used. The models perform poorly which could be the issue with the dataset itself.

1. The best adjusted R2 score achieved without any DR technique was 0.42.
2. After applying PCA, the number of features reduced from 7 to 5 and adjusted R2 improved to 0.44.
3. When other variants of PCA were tried, incremental PCA performed at par with PCA whereas sparse PCA increased dimensions instead.
4. When SVD is applied, the performance improves further to 0.47 whereas the number of features reduces from 7 to 5.

SVD proves to perform better for this dataset while reducing the number of features and improving adjusted R2 score.

5.2.4 Seoul Bike Sharing Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.569277709	0.572622742	422.9295	17	0.36367616	0.36600169	515.1175	8	0.393155154	0.39537295	503.0441	8	0.443229924	0.44577341	481.6216	10	0.414545898	0.41695297	493.9853	9
DecisionTreeRegressor	0.744458664	0.746443225	325.7616	17	0.586614042	0.58812482	415.1883	8	0.59469401	0.59617526	411.1106	8	0.71401106	0.71531754	345.1776	10	0.598905788	0.60055487	408.8752	9
ElasticNetCV	0.521784795	0.525498664	445.6367	17	0.468506409	0.47044882	470.7781	8	0.468527953	0.47047029	470.7685	8	0.46638436	0.46882207	471.5006	10	0.472827041	0.4749945	468.7531	9
GradientBoostingRegressor	0.839184431	0.840433342	258.4243	17	0.705383411	0.70646013	350.5063	8	0.702079661	0.70316845	352.466	8	0.811146335	0.81200907	280.4986	10	0.716250109	0.71741674	343.9026	9
KNeighborsRegressor	0.783313505	0.784996314	299.9751	17	0.759785666	0.76066356	316.4949	8	0.759054451	0.75993502	316.9762	8	0.725000827	0.7262571	338.4805	10	0.773882236	0.77481191	306.9976	9
XGBRegressor	0.87456752	0.875541641	228.2308	17	0.785574435	0.78635808	299.0236	8	0.782820857	0.78361457	300.9375	8	0.848540681	0.84923259	251.1979	10	0.78529454	0.78617729	299.1501	9
RandomForestRegressor	0.871547504	0.872545079	230.962	17	0.80493169	0.80564459	285.2072	8	0.805349763	0.80606114	284.9015	8	0.854770524	0.85543397	245.9775	10	0.804872519	0.80567478	285.1851	9
SVR	0.33432628	0.339495971	525.7751	17	0.287969759	0.29057197	544.8995	8	0.28791173	0.29051415	544.9217	8	0.287619289	0.29087365	544.7836	10	0.313852217	0.31667329	534.7816	9

Analysis:

For the bikes sharing dataset, adjusted R2 score is used to compare the results:

1. The best Adjusted R2 score was achieved to be 0.87 with 17 original features without any DR Technique.

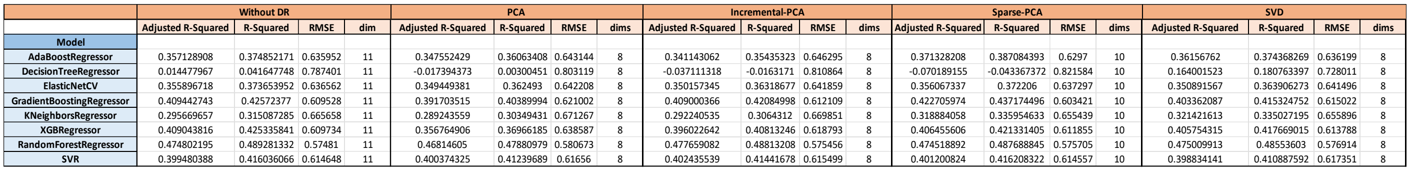
2. After applying PCA, the number of features reduced to 8 from 17 with some compromise on variance capture giving an R2 score of 0.81.

3. For the PCA variants, incremental PCA performs like PCA and sparse PCA performs better as it reduces the number of features from 17 to 10 and still captures the same amount of variance giving an adjusted R2 score of 0.86.

4. SVD performs at par with PCA however the number of dimensions is 1 more than PCA.

Sparse PCA performs well in this dataset as it reduces the number of features without any compromise on variance capture!

5.2.5 Red Wine Quality Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.357128908	0.374852171	0.635952	11	0.347552429	0.36063408	0.643144	8	0.341143062	0.35435323	0.646295	8	0.371328208	0.387084393	0.6297	10	0.36156762	0.374368269	0.636199	8
DecisionTreeRegressor	0.014477967	0.041647748	0.787401	11	-0.017394373	0.00300451	0.803119	8	-0.037111318	-0.0163171	0.810864	8	-0.070189155	-0.043367372	0.821584	10	0.164001523	0.180763397	0.728011	8
ElasticNetCV	0.355896718	0.373653952	0.636562	11	0.349449381	0.362493	0.642208	8	0.350157345	0.36318677	0.641859	8	0.356067337	0.372206	0.637297	10	0.350891567	0.363906273	0.641496	8
GradientBoostingRegressor	0.409442743	0.42572377	0.609528	11	0.391703515	0.40389994	0.621002	8	0.409000366	0.42084998	0.612109	8	0.422705974	0.437174496	0.603421	10	0.403362087	0.415324752	0.615022	8
KNeighborsRegressor	0.295669657	0.315087285	0.665658	11	0.289243559	0.30349431	0.671267	8	0.292240535	0.3064312	0.669851	8	0.318884058	0.335954633	0.655439	10	0.321421613	0.335027195	0.655896	8
XGBRegressor	0.409043816	0.425335841	0.609734	11	0.356764906	0.36966185	0.638587	8	0.396022642	0.40813246	0.618793	8	0.406455606	0.421331405	0.611855	10	0.405754315	0.417669015	0.613788	8
RandomForestRegressor	0.474802195	0.489281332	0.57481	11	0.46814605	0.47880979	0.580673	8	0.477659082	0.48813208	0.575456	8	0.474518892	0.487688845	0.575705	10	0.475009913	0.48553603	0.576914	8
SVR	0.399480388	0.416036066	0.614648	11	0.400374325	0.41239689	0.61656	8	0.402435539	0.41441678	0.615499	8	0.401200824	0.416208322	0.614557	10	0.398834141	0.410887592	0.617351	8

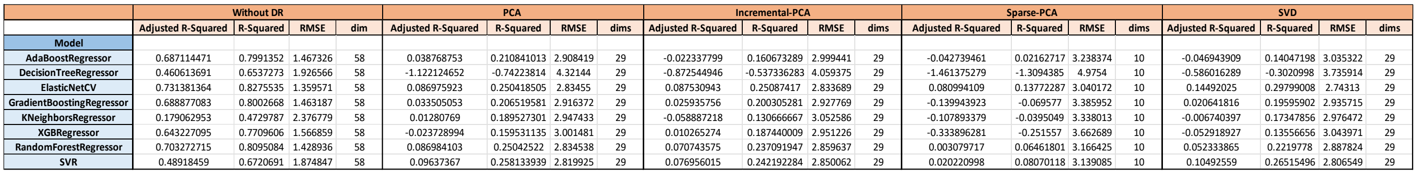
Analysis:

For the Red wine quality dataset, adjusted R2 score is used to compare the results:

1. The best Adjusted R2 score was achieved to be 0.47 with 11 original features without any DR Technique.
2. After applying PCA, the number of features reduced to 8 from 11 with no difference in R2 score proving it to be a particularly good DR technique for this dataset
3. For the PCA variants, both incremental and sparse performed at par with PCA
4. SVD performs reduces number of features to 8 from 11 which is at par with PCA however the adjusted R2 slightly improves but 0.01 difference is not generalizable.

PCA, its variants and SVD all perform well for this dataset!

5.2.6 Student Portuguese Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.687114471	0.7991352	1.467326	58	0.038768753	0.210841013	2.908419	29	-0.022337799	0.160673289	2.999441	29	-0.042739461	0.02162717	3.238374	10	-0.046943909	0.14047198	3.035322	29
DecisionTreeRegressor	0.460613691	0.6537273	1.926566	58	-1.122124652	-0.74223814	4.32144	29	-0.872544946	-0.537336283	4.059375	29	-1.461375279	-1.3094385	4.9754	10	-0.586016289	-0.3020998	3.735914	29
ElasticNetCV	0.731381364	0.8275535	1.359571	58	0.086975923	0.250418505	2.83455	29	0.087530943	0.25087417	2.833689	29	0.080994109	0.13772287	3.040172	10	0.14492025	0.29799008	2.74313	29
GradientBoostingRegressor	0.688877083	0.8002668	1.463187	58	0.033505053	0.206519581	2.916372	29	0.025935756	0.200305281	2.927769	29	-0.139943923	-0.069577	3.385952	10	0.020641816	0.19595902	2.935715	29
KNeighborsRegressor	0.179062953	0.4729787	2.376779	58	0.01280769	0.189527301	2.947433	29	-0.058887218	0.130666667	3.052586	29	-0.107893379	-0.0395049	3.338013	10	-0.006740397	0.17347856	2.976472	29
XGBRegressor	0.643227095	0.7709606	1.566859	58	-0.023728994	0.159531135	3.001481	29	0.010265274	0.187440009	2.951226	29	-0.333896281	-0.251557	3.662689	10	-0.052918927	0.13556656	3.043971	29
RandomForestRegressor	0.703272715	0.8095084	1.428936	58	0.086984103	0.25042522	2.834538	29	0.070743575	0.237091947	2.859637	29	0.003079717	0.06461801	3.166425	10	0.052333865	0.2219778	2.887824	29
SVR	0.48918459	0.6720691	1.874847	58	0.09637367	0.258133939	2.819925	29	0.076956015	0.242192284	2.850062	29	0.020220998	0.08070118	3.139085	10	0.10492559	0.26515496	2.806549	29

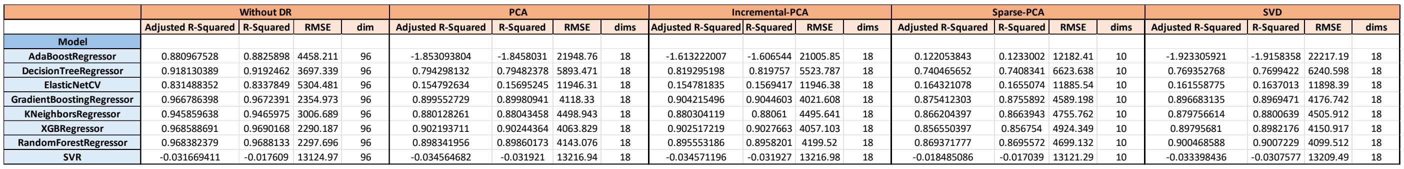
Analysis:

Adjusted R2 score is used to compare the results for the student performance dataset:

1. When no DR technique was applied, the best Adjusted R2 score achieved was 0.70.
2. When PCA is applied, the features reduced from 58 to 29 but the best the adjusted R2 score drops significantly which means PCA is not a good DR technique for this dataset.
3. The rest of the PCA variants and SVD performs poorly as well!

This dataset does not seem to allow any feature reduction which could be highlighting that there is a remarkably high correlation between some variables with predicted value!

5.2.7 Tom’s Hardware Dataset



	Without DR				PCA				Incremental-PCA				Sparse-PCA				SVD			
	Adjusted R-Squared	R-Squared	RMSE	dim	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims	Adjusted R-Squared	R-Squared	RMSE	dims
Model																				
AdaBoostRegressor	0.880967528	0.8825898	4458.211	96	-1.853093804	-1.8458031	21948.76	18	-1.613222007	-1.606544	21005.85	18	0.122053843	0.1233002	12182.41	10	-1.923305921	-1.9158358	22217.19	18
DecisionTreeRegressor	0.918130389	0.9192462	3697.339	96	0.794298132	0.79482378	5893.471	18	0.819295198	0.819757	5523.787	18	0.740465652	0.7408341	6623.638	10	0.769352768	0.7699422	6240.598	18
ElasticNetCV	0.831488352	0.8337849	5304.481	96	0.154792634	0.15695245	11946.31	18	0.154781835	0.1569417	11946.38	18	0.164321078	0.1655074	11885.54	10	0.161558775	0.1637013	11898.39	18
GradientBoostingRegressor	0.966786398	0.9672391	2354.973	96	0.899552729	0.89980941	4118.33	18	0.904215496	0.9044603	4021.608	18	0.875412303	0.8755892	4589.198	10	0.896683135	0.8969471	4176.742	18
KNeighborsRegressor	0.945859638	0.9465975	3006.689	96	0.880128261	0.88043458	4498.943	18	0.880304119	0.88061	4495.641	18	0.866204397	0.8663943	4755.762	10	0.879756614	0.8800639	4505.912	18
XGBRegressor	0.968588691	0.9690168	2290.187	96	0.902193711	0.90244364	4063.829	18	0.902517219	0.9027663	4057.103	18	0.856550397	0.856754	4924.349	10	0.89795681	0.8982176	4150.917	18
RandomForestRegressor	0.968382379	0.9688133	2297.696	96	0.898341956	0.89860173	4143.076	18	0.895553186	0.8958201	4199.52	18	0.869371777	0.8695572	4699.132	10	0.900468588	0.9007229	4099.512	18
SVR	-0.031669411	-0.017609	13124.97	96	-0.034564682	-0.031921	13216.94	18	-0.034571196	-0.031927	13216.98	18	-0.018485086	-0.017039	13121.29	10	-0.033398436	-0.0307577	13209.49	18

Analysis:

Adjusted R2 score is used to compare the results for the Tom's Hardware dataset:

1. When no DR technique was applied, the best Adjusted R2 score achieved was 0.97 which means the model captures almost all the variance available.
2. When PCA is applied, the features reduced from 96 to 18 but with a small drop in adjusted R2 score of 0.06 which leads to an R2 score of 0.90.
3. As other PCA variants are tried, incremental PCA performs at par with PCA but sparse reduces dimensions to 10 with some more compromise on adjusted R2 value leading to R2 score of 0.87
4. SVD performs at par with PCA as it leads to an adjusted R2 score of 0.9 and gives a feature set of 18 dimensions.

PCA and SVD both perform well as they reduce the number of features from 96 to 18 and capture all variance!

6. Critical Analysis

All the datasets do not compromise a lot on variance capture when their dimensionality is reduced using any of the techniques tried above which proves that these techniques are especially useful and should be implemented before diving deep into machine learning. Spending some time on reducing dimensions would help in the long run since model development becomes easier and less time-consuming when dataset features are reduced. Furthermore, LDA is only applicable on classification datasets and there are other varying factors which does not allow generalizability amongst all datasets however, it can be concurred that PCA works well for all datasets as it helps reduce dimensions more significantly and does not compromise a lot on variance capture which is the major goal of PCA itself. The major reason that these techniques work so well is because tabular data can be explained via linear mappings. Since PCA, LDA and SVD all are linear transformers, they can capture the hidden trends in the data well. However, same cannot be said for textual or image data where there are non-linear patterns.

6.1 Classification Datasets

Most of the techniques performed well but Linear discriminant analysis stood out for every dataset. LDA significantly reduced dimensions without compromising on variance capture making it an extremely useful technique. The major reason LDA can outperform the other techniques is because it is a supervised one which could be considered its drawback as well. However, since we are focused on classification, we would require a labeled dataset and LDA uses the labels along with the dataset to increase separability between classes. PCA only focuses on the linear mappings between the predictor variable whereas LDA focuses on linear mapping between the entire dataset and predicted variable as well. In addition, LDA's focus on class separability becomes the major factor that helps in improving classification since LDA focuses on capturing distinct information between the classes. However, LDA forcefully reduces dimensions to less than number of classes which may cause losing essential information if the entire dataset is relevant so we must resort to PCA or SVD if there is a major performance drop with LDA.

6.2 Regression Datasets

For regression datasets, LDA was not applicable as it requires a classification dataset. PCA worked well for these as it was able to capture most of the variance with significant feature reduction. Singular value decomposition performed like PCA as well where in some cases it was able to surpass PCA however PCA still remained the winner technique for regression datasets. The reason PCA works so well is because it can remove multicollinearity between variables and map the dataset on independent dimensions that capture the highest variance. Furthermore, because regression has a continuous output, it is more prone to having noise than classification datasets so amongst the other techniques PCA can handle noise better.

7. Conclusion

Overall, DR techniques must become a standard pre-processing step for high dimensional datasets to avoid unnecessary prolonged computation time and make machine learning simpler.

8. References

Chaitanyanarava. “A Complete Guide on Dimensionality Reduction.” *Medium*, Analytics Vidhya, 18 Apr. 2020, https://medium.com/analytics-vidhya/a-complete-guide-on-dimensionality-reduction-62d9698013d2.

Medium's Distribution Standards: What Writers and ... - Medium Help Center. https://help.medium.com/hc/en-us/articles/360006362473-Medium-s-Distribution-Standards-What-Writers-and-Publications-Need-to-Know.

Xiaozhou, YANG. “Linear Discriminant Analysis, Explained.” *Medium*, Towards Data Science, 27 Jan. 2022, https://towardsdatascience.com/linear-discriminant-analysis-explained-f88be6c1e00b.

Lever, Jake, et al. “Principal Component Analysis.” *Nature News*, Nature Publishing Group, 29 June 2017, https://www.nature.com/articles/nmeth.4346#:~:text=PCA%20helps%20you%20interpret%20your,act%20as%20summaries%20of%20features.

“6.5. Unsupervised Dimensionality Reduction.” *Scikit*, https://scikit-learn.org/stable/modules/unsupervised_reduction.html.

Desai, Rashi. “How to Start Writing Data Science Blogs?” *Medium*, Towards Data Science, 4 June 2020, https://towardsdatascience.com/how-to-start-writing-data-science-blogs-73bc55f59169.

Rajamani, Ranjani. “High on Accuracy but Low on ROC Score?” *Medium*, Medium, 16 Mar. 2021, https://ranjani-rajamani.medium.com/high-on-accuracy-but-low-on-roc-score-a40f2053b6c4.