DIMENSIONALITY REDUCTION USING DISCRETE COSINE TRANSFORM(DCT) FOR HIGH DIMENSIONAL DATA

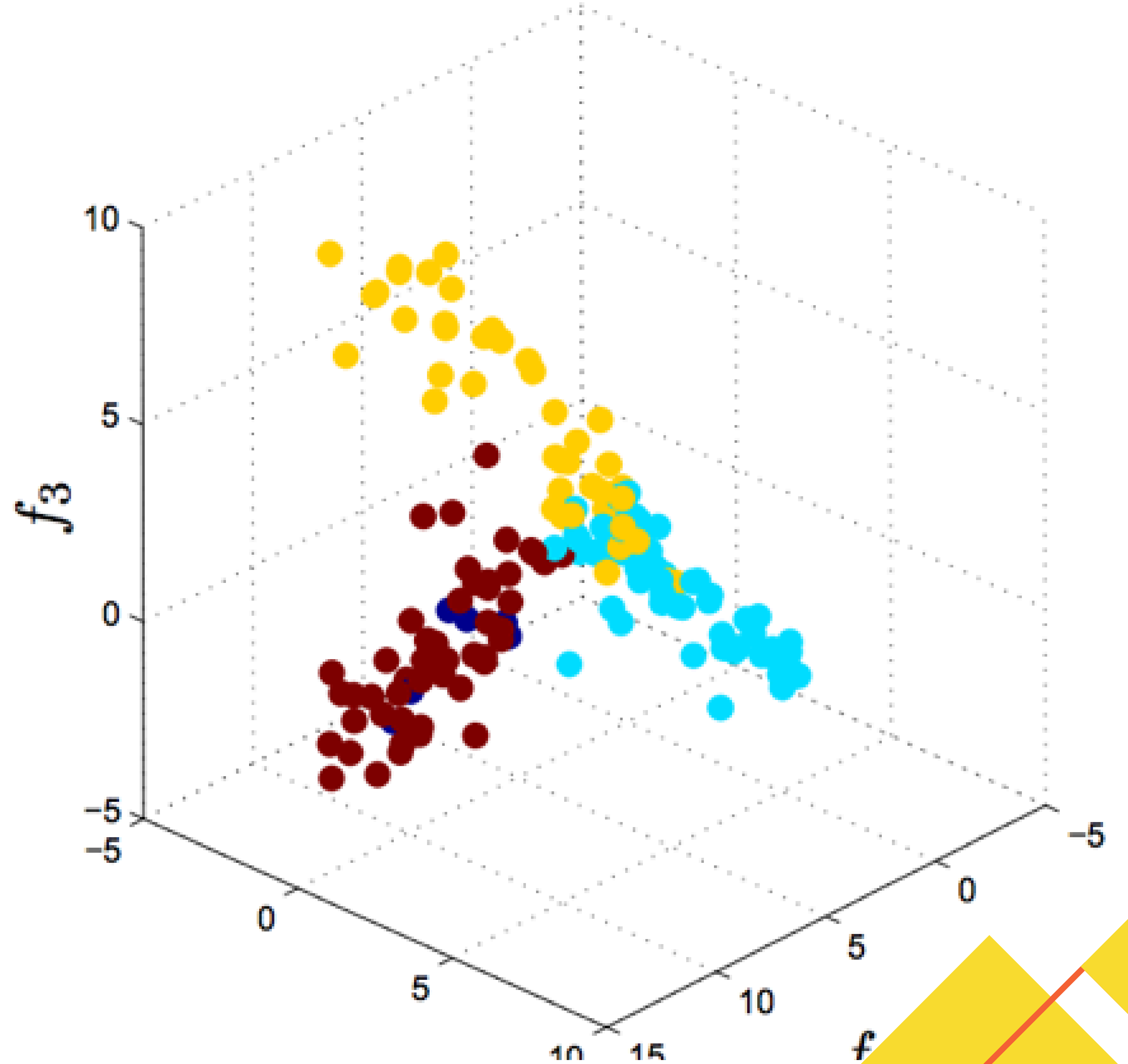
AVNISH RAJ (25/SWE/03)
PRAJJWAL PATHAK (25/SWE/05)
SUMIT TRIPATHI (25/SWE/08)
NITESH BHARDWAJ (25/SWE/12)

WHY WE NEED DIMENSIONALITY REDUCTION

- High-dimensional data increases computational cost.
- Causes the "curse of dimensionality" — models become less efficient.
- Leads to redundant and noisy features in datasets.
- Makes visualization and interpretation difficult.
- Can result in overfitting due to too many features.
- Dimensionality reduction helps in faster, simpler, and more accurate modelling.

PRINCIPAL COMPONENT ANALYSIS (PCA)

- A linear dimensionality reduction technique.
- Transforms correlated features into uncorrelated principal components.
- Each principal component captures maximum variance in the data.
- Helps represent data in a lower-dimensional space.
- Commonly used for feature extraction, noise reduction, and visualization.



PRINCIPAL COMPONENT ANALYSIS (PCA)

ADVANTAGES

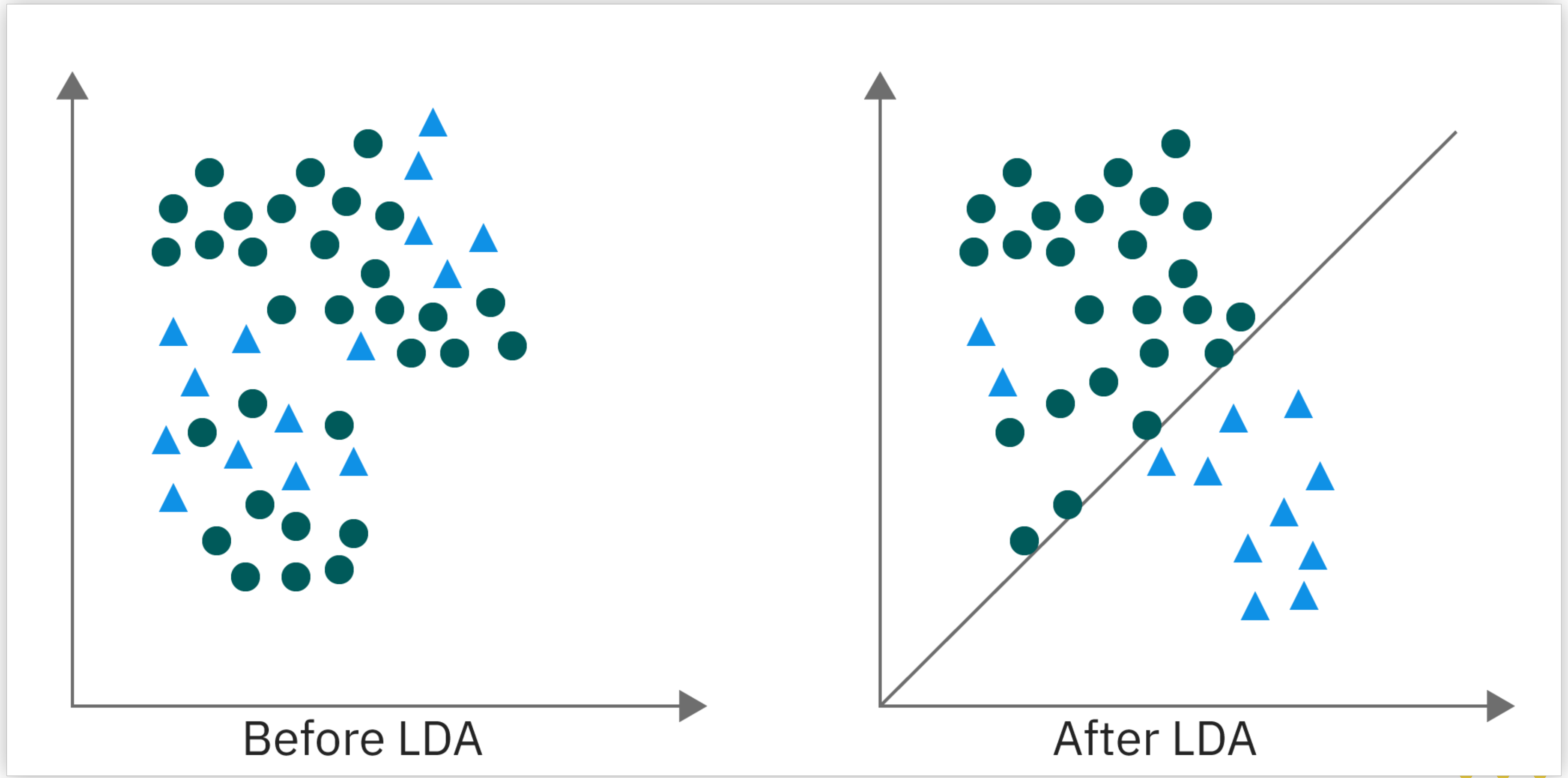
- Reduces data size without major information loss.
- Removes multicollinearity between features.
- Improves model performance and training speed.
- Helps visualize high-dimensional data in 2D or 3D.

DISADVANTAGES

- Interpretation of principal components is difficult.
- Assumes linear relationships only.
- Scaling sensitive — results depend on data normalization.
- May lose important small-variance features.

LINEAR DISCRIMINANT ANALYSIS (LDA)

- A supervised dimensionality reduction technique.
- Projects data onto a lower-dimensional space while maximizing class separability.
- Works by finding linear combinations of features that best distinguish classes.
- Commonly used for classification and pattern recognition tasks.
- Example applications: face recognition, medical diagnosis, speech recognition.



LINEAR DISCRIMINANT ANALYSIS (LDA)

ADVANTAGES

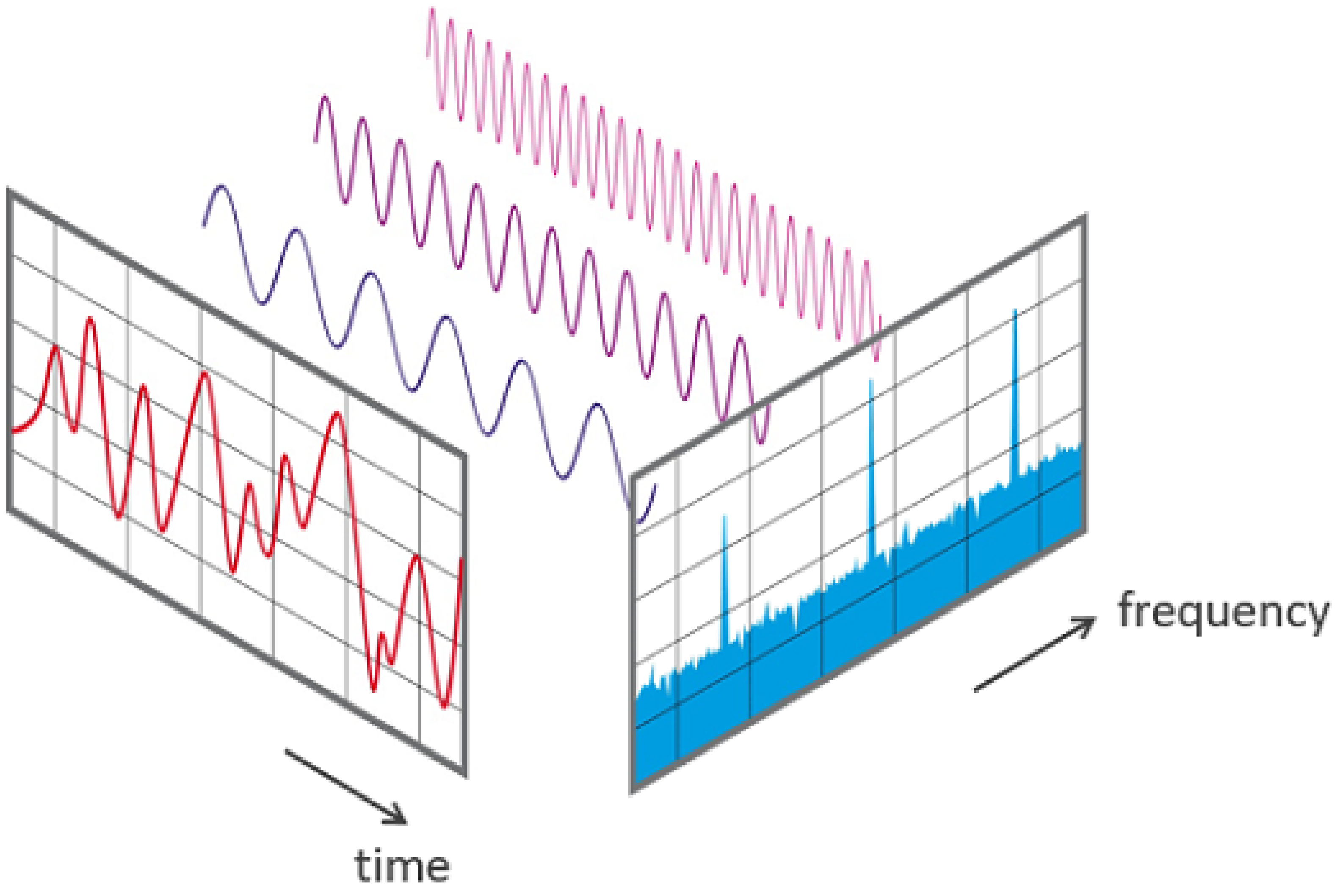
- Enhances class discrimination between groups.
- Reduces dimensionality while preserving class information.
- Works well when classes are linearly separable.
- Improves performance of classifiers like SVM or Logistic Regression.

DISADVANTAGES

- Assumes normal distribution of features.
- Sensitive to outliers and non-linear class boundaries.
- Requires labeled data for training.
- Can perform poorly with overlapping or unbalanced classes.

WHAT IS DISCRETE COSINE TRANSFORM (DCT)?

- DCT converts spatial or time-domain data into frequency-domain components.
- Represents data as a sum of cosine functions with varying frequencies and amplitudes.
- Useful for data compression, noise reduction, and dimensionality reduction.
- Most of the data's energy is concentrated in a few low-frequency coefficients.



MATHEMATICAL FOUNDATION OF DCT

- DCT is a linear orthogonal transformation — preserves total signal energy.
- Can be viewed as projecting data onto cosine basis functions.
- Type-II DCT (and its inverse, DCT-III) are most widely used.
- The transformation ensures real-valued, symmetric basis functions — ideal for data compression.

$$X[k] = \sum_{n=0}^{N-1} x[n] \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)k\right]$$

(Time to frequency domain)

$$x[n] = \frac{1}{2}X[0] + \sum_{k=1}^{N-1} X[k] \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)k\right]$$

(Frequency domain to Time domain)

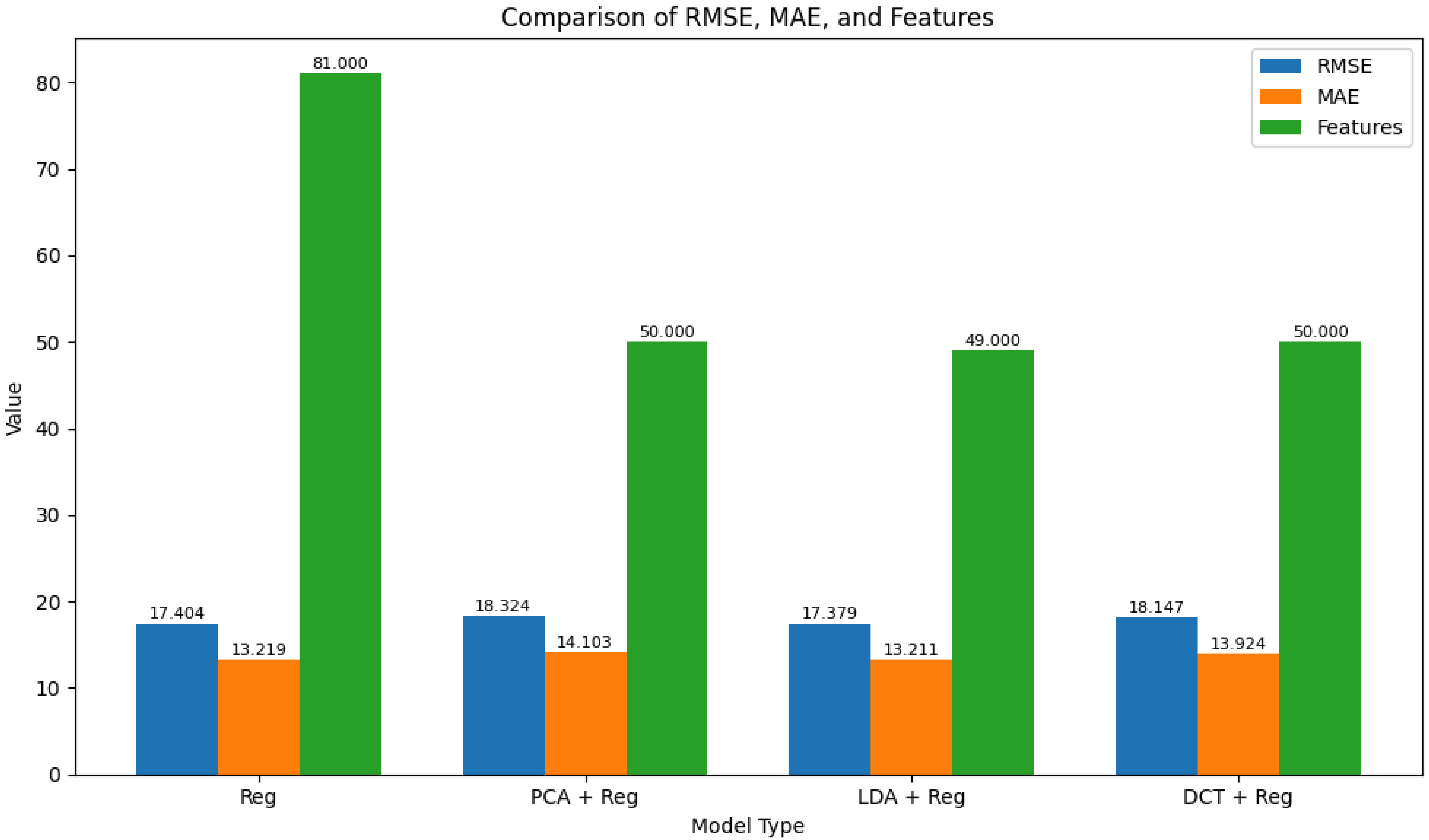
WHY DCT WORKS FOR DATA REDUCTION

- DCT coefficients show how much energy (information) lies in each frequency component.
- Low-frequency coefficients carry most of the useful data;
- high-frequency coefficients represent fine details or noise.
- By keeping only the first few coefficients, we achieve:
    - High compression ratio
    - Low reconstruction error
    - Faster computation for ML models

METHODOLOGY

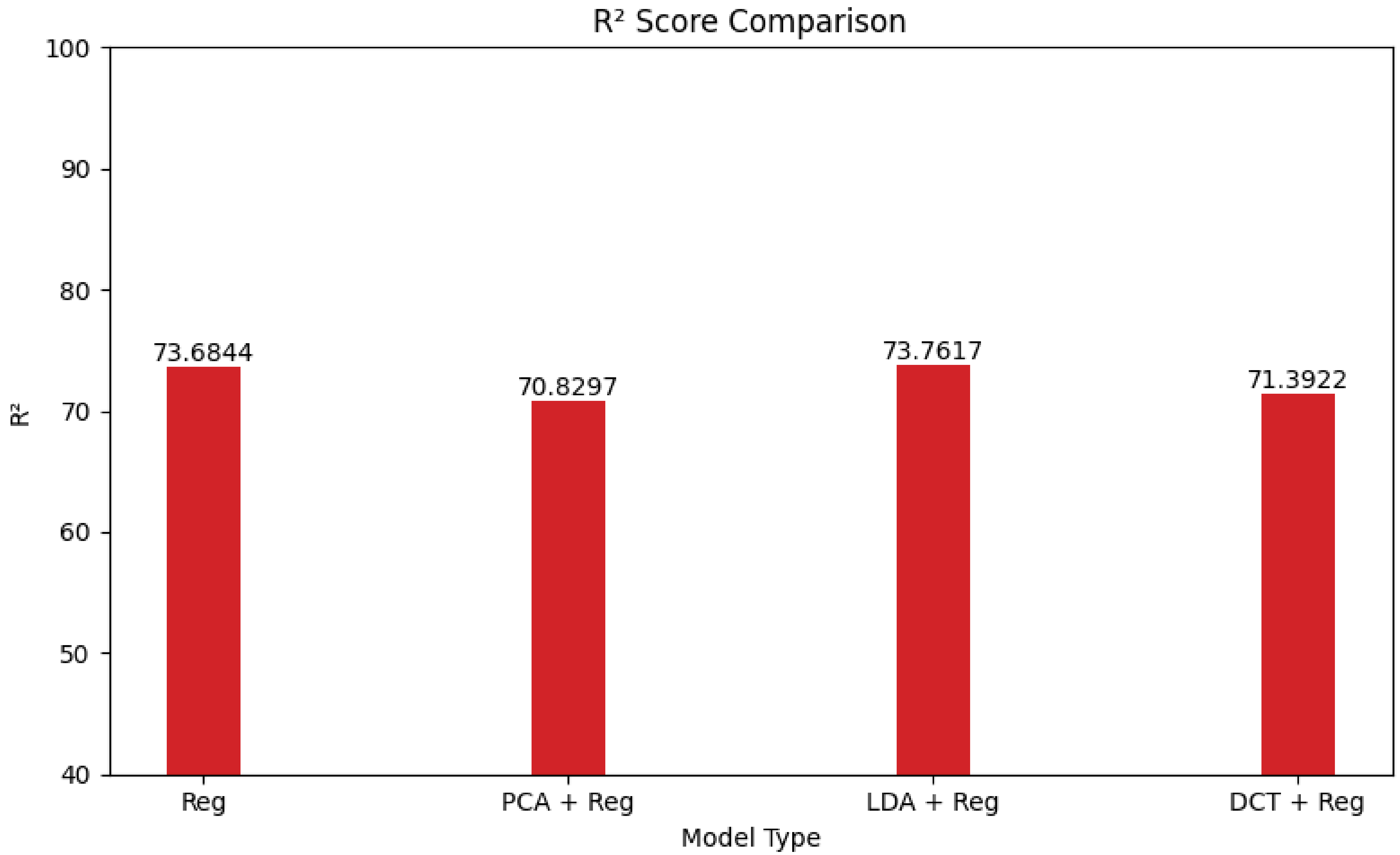
1. **Data load & split** — read CSV, set target, split into train/test.
2. **Preprocessing** — mean imputation for missing values; standard scaling (fit on train only).
3. **Baseline model** — train Linear regression on full (scaled) features for reference.
4. **PCA pipeline** — fit PCA on training features, keep top k components, train Linear regression .
5. **LDA pipeline** — bin continuous target into quantile classes, fit LDA (n_classes-1 components), train Linear regression .
6. **DCT pipeline** — apply 1-D DCT across each sample (axis=1), keep first k coefficients (energy compaction), train Linear regression .
7. **Evaluation & diagnostics** — compute RMSE, MAE, R^2, record training time; extra metrics: PCA explained variance, DCT energy retained.
8. **Compare & visualize** — assemble metrics table and plot comparisons (RMSE, MAE, R^2, training time, feature counts).

RESULTS



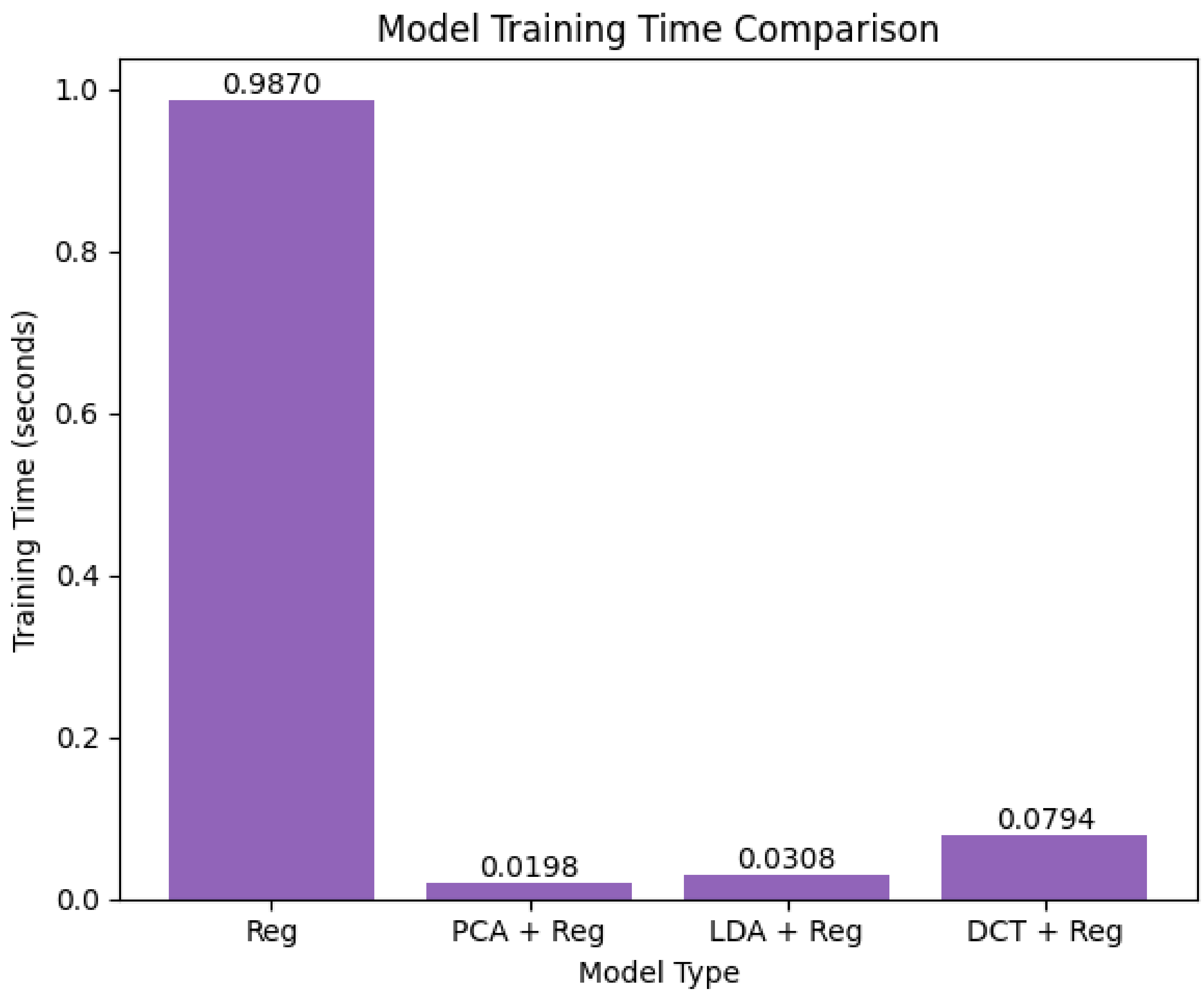
RMSE and MAE comparision of PCA, LDA, DCT

RESULTS



R2 comparision of PCA, LDA, DCT

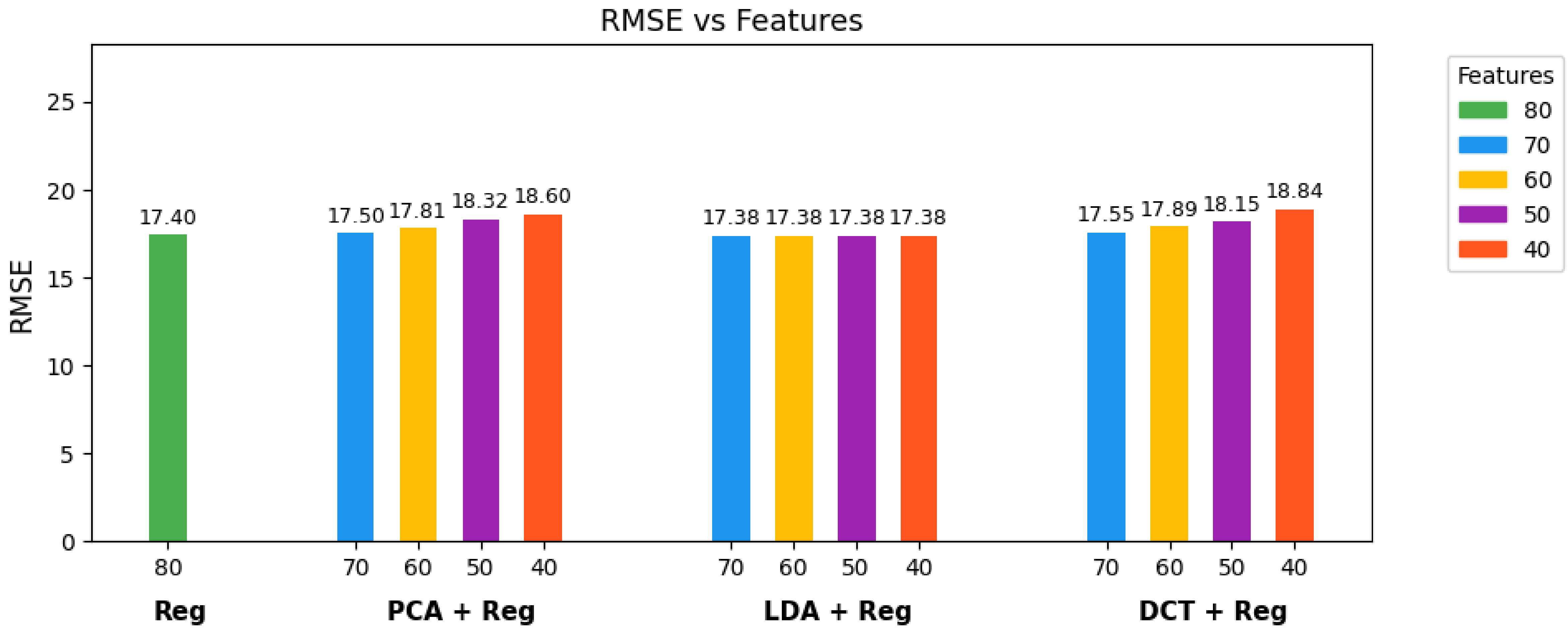
RESULTS

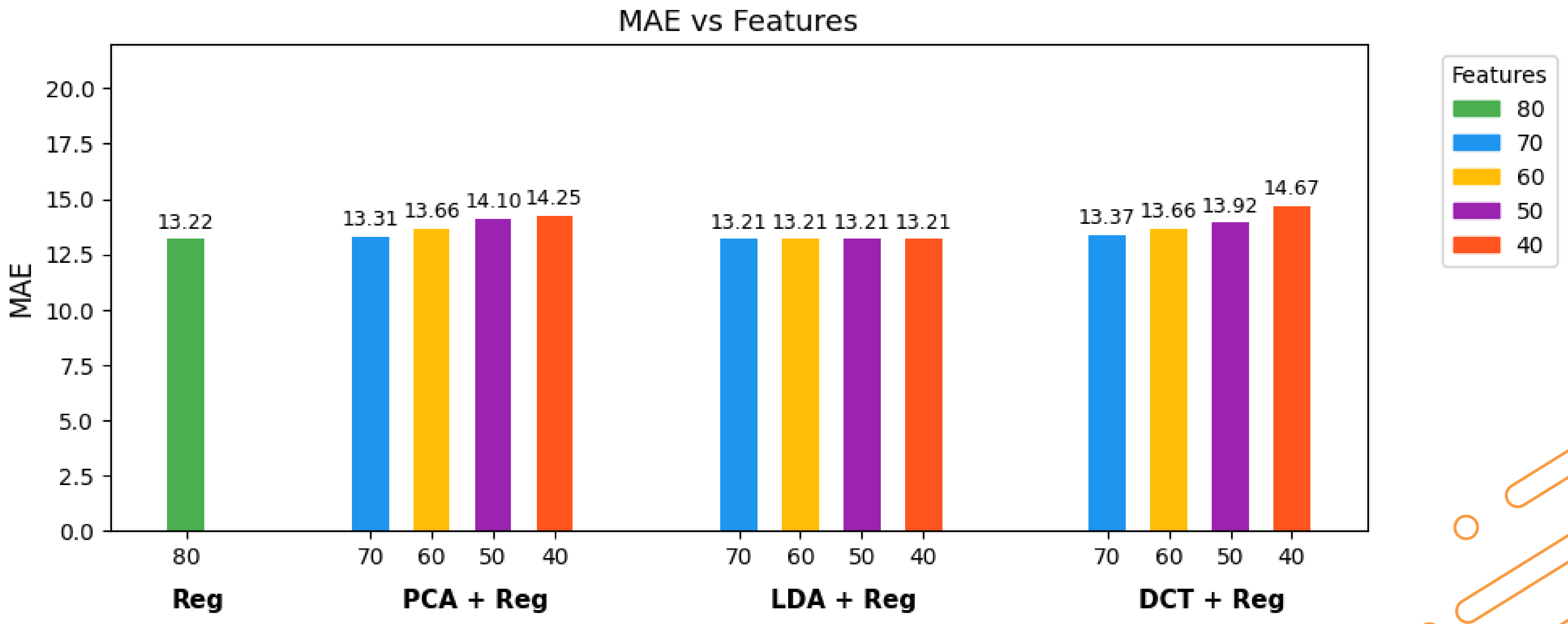


Model Training Time Comparison
Training Time (seconds)
1.0
0.8
0.6
0.4
0.2
0.0
0.9870
0.0198
0.0308
0.0794
Reg
PCA + Reg
LDA + Reg
DCT + Reg
Model Type

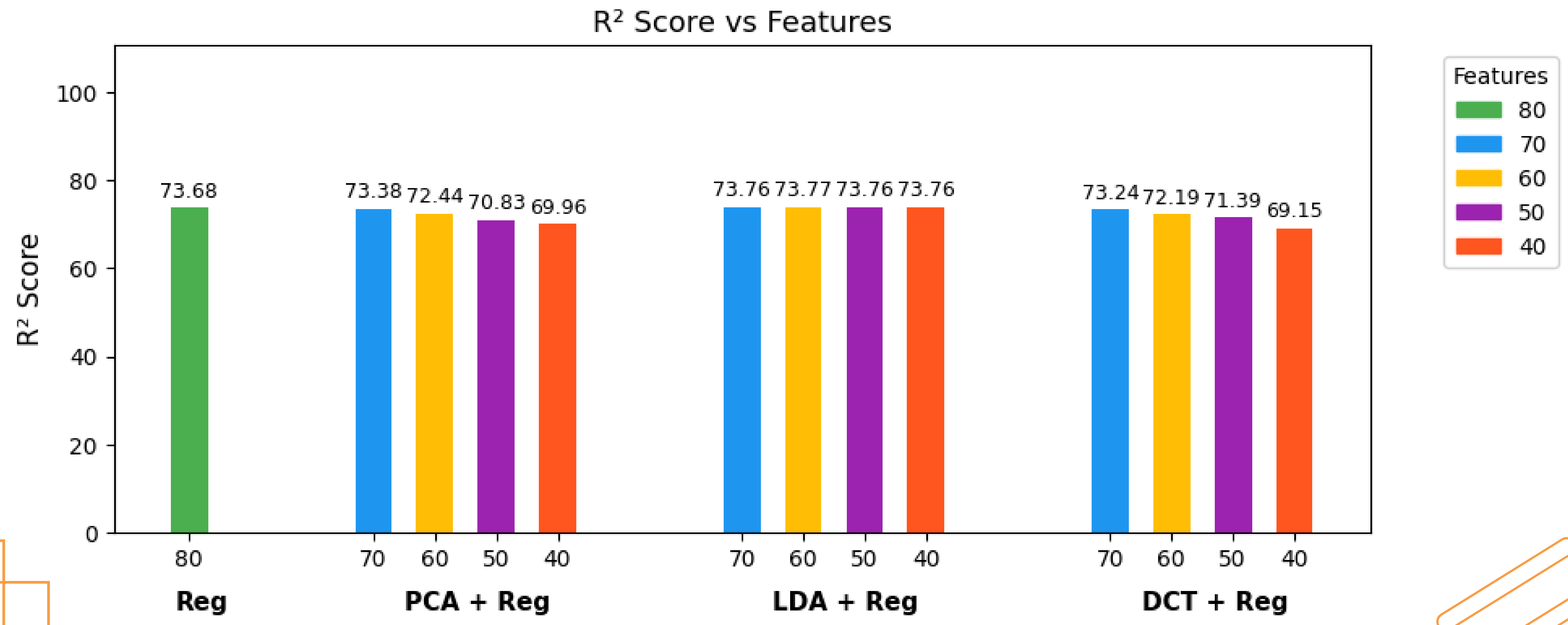
RESULTS

RMSE and MAE comparison of PCA, LDA, DCT with different number of features



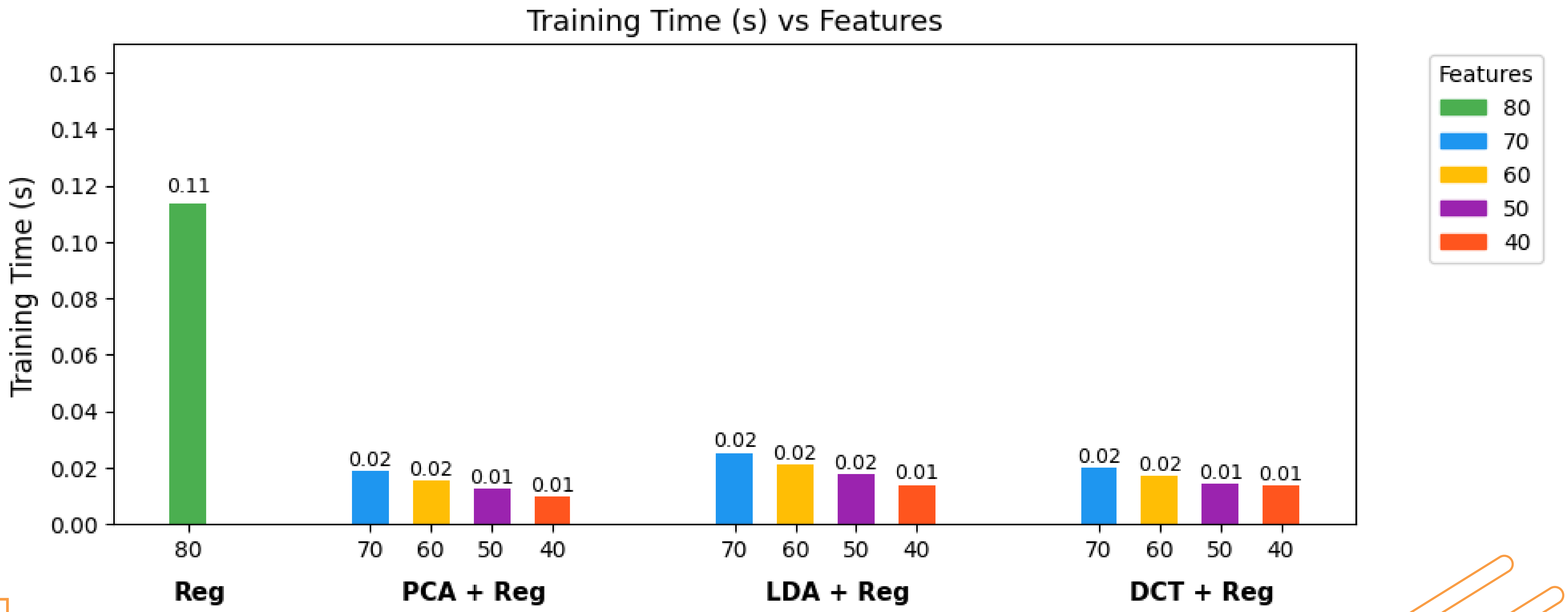


RESULTS



R2 comparison of PCA, LDA, DCT with different number of features

RESULTS



Training time comparison of PCA, LDA, DCT with different number of features

CONCLUSION

- DCT-based dimensionality reduction effectively compressed high-dimensional data while preserving essential information.
- The DCT + Linear Regression pipeline achieved competitive accuracy with reduced computation time.
- The hybrid system offered faster training, noise resilience, and simpler implementation.
- Confirms that frequency-domain transformation can serve as a powerful, interpretable alternative to classical reduction methods.

THANK YOU