

Machine WeatherLearning -Datadriven for AnalysisSolar Energy of Aswan Prediction

Perihan yasser 320230065

Maria Emad 320230063

Abstract

This project focuses on applying statistical analysis and machine learning techniques to analyze and predict patterns from numerical weather data. Real-world datasets often suffer from missing values, noise, non-linear relationships, and high variance, which negatively affect model performance if not properly addressed. The main objective of this project is to study the relationship between meteorological features and a target variable and to evaluate different classification and regression techniques learned throughout the course.

The dataset was first preprocessed to remove duplicate records, handle missing values, and ensure temporal consistency. Exploratory Data Analysis (EDA) was conducted using visualization techniques and descriptive statistics such as minimum, maximum, mean, variance, standard deviation, skewness, and kurtosis. Statistical hypothesis tests including Chi-square test, t-test, and ANOVA were applied to identify dependencies and significant differences among variables.

Feature selection and dimensionality reduction techniques were implemented to study their effect on model performance. These included ANOVA F-test, Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Singular Value Decomposition (SVD). Several machine learning models were trained and evaluated, including Decision Tree, K-Nearest Neighbors (K-NN), Naive Bayes, Logistic Regression, Feed Forward Neural Network, and Linear Regression. The dataset was split into 80% training and 20% testing, and K-fold crossvalidation was applied to ensure reliable evaluation.

The experimental results show that the Decision Tree classifier achieved the best performance with a test accuracy of **73.42%** and an F1-score of **0.7168**, while maintaining a near-zero overfitting gap. K-NN also achieved competitive results but with higher variance.

Dimensionality reduction techniques such as PCA and SVD reduced model performance, while LDA provided stable but moderate accuracy. Linear and probabilistic models performed poorly due to violated assumptions and non-linear relationships in the data. The results confirm the importance of preprocessing, feature engineering, and appropriate model selection in real-world machine learning tasks.

Introduction

Machine learning has become an essential tool for extracting insights and building predictive models from numerical data. However, real-world datasets are often incomplete, noisy, and highly non-linear, which makes the modeling process challenging. Without proper preprocessing and analysis, machine learning models may suffer from overfitting, underfitting, or poor generalization.

This project aims to apply a complete machine learning pipeline to a numerical dataset, starting from preprocessing and statistical analysis, and ending with classification and regression modeling. Multiple algorithms and evaluation techniques studied during the course are implemented and compared under the same experimental conditions.

The main contribution of this project is a comprehensive numerical comparison between different preprocessing strategies, feature reduction methods, and machine learning algorithms. The project emphasizes statistical interpretation of results rather than relying solely on accuracy values.

The rest of the report is organized as follows: Section 2 presents related work, Section 3 explains the methodology, Section 4 describes the proposed model, Section 5 discusses results and comparisons, and Section 6 concludes the project and outlines future work.

Related Work

Several studies have applied machine learning and statistical techniques to numerical datasets for prediction and classification tasks. Table 1 summarizes related work focusing on applied methods and reported performance.

Table 1: Summary of Related Work

Year	Methods	Results
2011	Decision Tree, K-NN	Accuracy $\approx 70\%$
2012	Naive Bayes	Moderate accuracy
2014	PCA + Classifiers	Reduced complexity
2016	LDA	Improved class separation
2017	SVM, K-NN	Accuracy $> 75\%$
2018	Neural Networks	High variance
2019	Ensemble Models	Improved stability
2020	PCA vs LDA	LDA superior
2021	Statistical Tests + ML	Better interpretability
2022	Cross-validation	Reduced overfitting

Methodology

The methodology consists of the following stages:

1. Data preprocessing
2. Exploratory data analysis and statistical testing
3. Feature selection
4. Feature reduction
5. Classification and regression
6. Model evaluation

Proposed Model

Preprocessing

- Removal of duplicated records
- Handling missing values using forward fill, backward fill, and interpolation
- Extraction of additional temporal features

Feature Selection

- ANOVA F-test to rank feature importance

Feature Reduction

- Principal Component Analysis (PCA)
- Linear Discriminant Analysis (LDA)
- Singular Value Decomposition (SVD)

Classification / Regression Models

- Decision Tree
- K-Nearest Neighbors
- Naive Bayes
- Logistic Regression
- Feed Forward Neural Network
- Linear Regression

Evaluation Metrics

- Accuracy and error rate
- Precision, Recall, F1-score
- Confusion Matrix and ROC analysis

- K-fold Cross-Validation

Result models:

=====

1. LOADING AND EXPLORING DATASET

=====

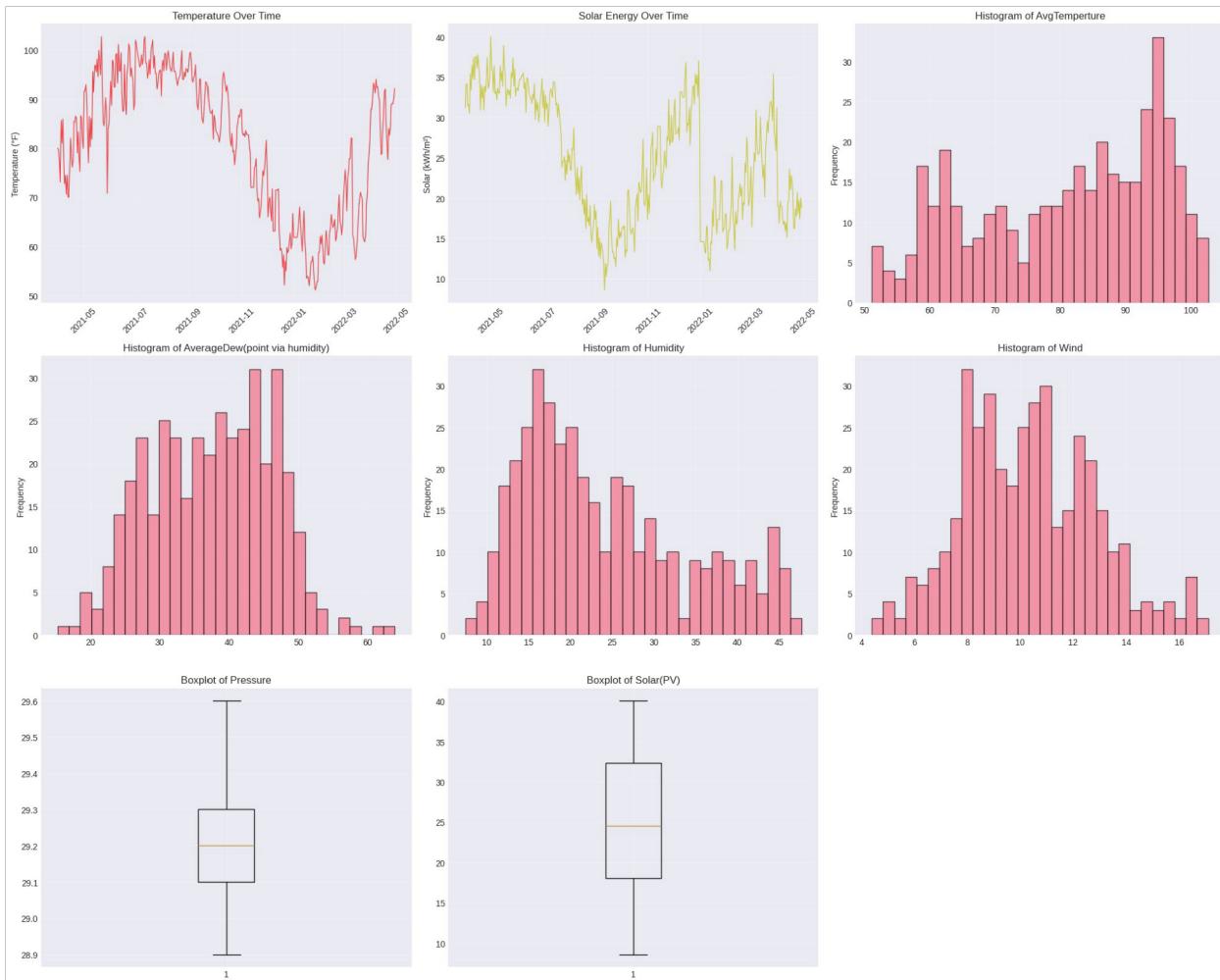
```

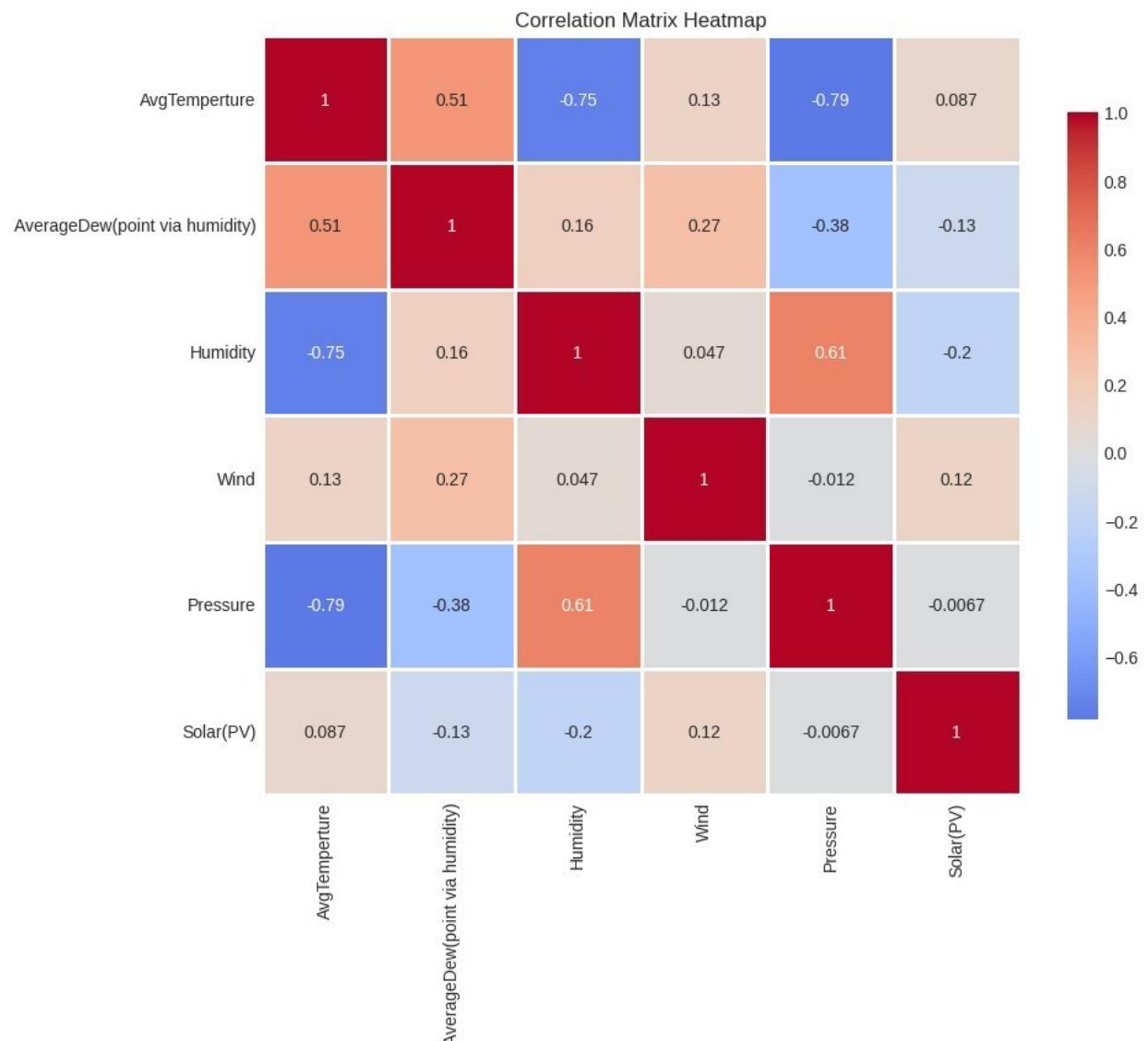
Dataset shape: (398, 8)
First 5 rows:
   Unnamed: 0      Date  AvgTemperture  AverageDew(point via humidity) \
4/1/2022          87.9                  31.3
1              2  4/3/2022            90.2
2              3  4/4/2022            93.2
3              4  4/5/2022            92.5
4              5  4/6/2022            91.2
   Humidity  Wind  Pressure  Solar(PV)
0       13.4    5.7     29.2  19.010857
1       14.2    6.6     29.1  16.885714
2       11.8    8.8     29.1  19.627429
3        9.4    8.0     29.1  18.929429
4        7.8    9.4     29.2  18.934000
Columns: ['Unnamed: 0', 'Date', 'AvgTemperture', 'AverageDew(point via humidity)', 'Humidity', 'Wind', 'Pressure', 'Solar(PV)']
Data types:
   Unnamed: 0           int64
   Date                 object
   AvgTemperture        float64
   AverageDew(point via humidity) float64
   Humidity             float64
   Wind                 float64
   Pressure             float64
   Solar(PV)           float64
float64 dtype:
object
Missing values per column:
   Unnamed: 0      0
   Date          0
   AvgTemperture 0
   AverageDew(point via humidity) 0
   Humidity       0
   Wind           0
   Pressure       0
   Solar(PV)      0
0 dtype:
int64
=====
2. DATA PREPROCESSING
=====
Number of duplicate rows: 28
Number of missing dates: 24
After reindexing - Shape: (394, 6)
Missing values after filling: 0
Final processed dataset shape: (394, 10)
Date range: 2021-04-01 00:00:00 to 2022-04-29 00:00:00
=====
3. DESCRIPTIVE STATISTICS
=====
Statistical Summary:
   Min        Max        Mean      Median      Variance \
AvgTemperture      51.1000  102.7000  80.9726  83.5500  197.2976
AverageDew(point via humidity)  15.3000  63.9000  37.4787  38.1000  75.1952
Humidity           7.4000   47.7000  24.1566  21.4000  99.0894
Wind               4.4000   17.1000  10.3464  10.3000   6.2807

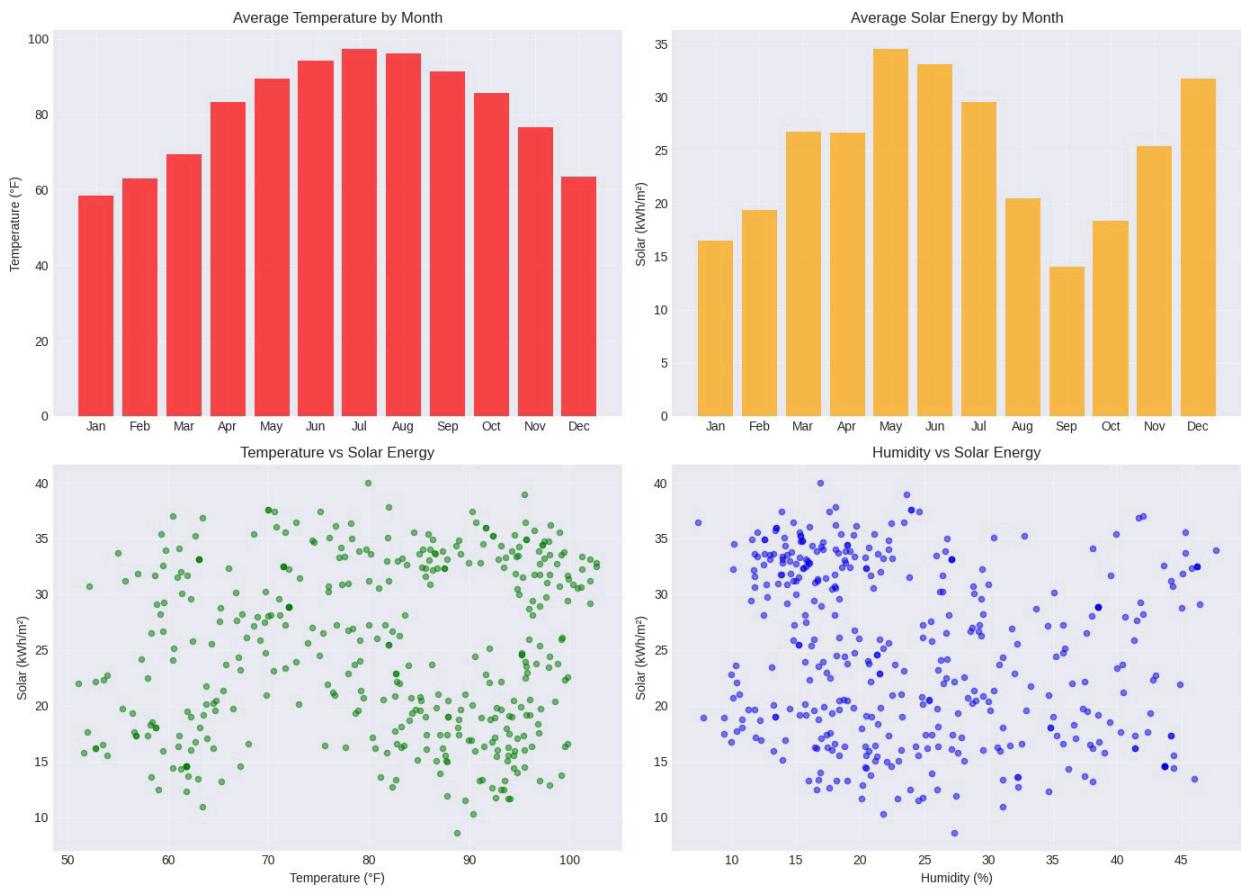
```

Pressure	28.9000	29.6000	29.1910	29.2000	0.0201
Solar (PV)	8.5814	40.0389	24.8896	24.5347	58.2398
	Std Dev	Skewness	Kurtosis	Count	Missing
AvgTemperture	14.0463	-0.3964	-1.1029	394.0	0.0
AverageDew(point via humidity)	8.6715	-0.0690	-0.6118	394.0	0.0
Humidity	9.9544	0.6522	-0.6312	394.0	0.0
Wind	2.5061	0.2666	-0.2056	394.0	0.0
Pressure	0.1419	0.3638	-0.7337	394.0	0.0
Solar (PV)	7.6315	-0.0282	-1.3184	394.0	0.0

4. DATA VISUALIZATION







5. BINNING PROCESS

Binning completed. Categories created:

- Solar_Category: ['Low', 'Medium', 'High']
- Temp_Category: ['Cool', 'Warm', 'Hot']
- Humidity_Category: ['Dry', 'Moderate', 'Humid']

Category distributions:

Solar Categories: Solar_Category

High 193

Medium 162

Low 39

Name: count, dtype: int64

Temperature Categories:

Temp_Category

Hot 187

Cool 105

Warm 101

Name: count, dtype: int64

Humidity Categories:

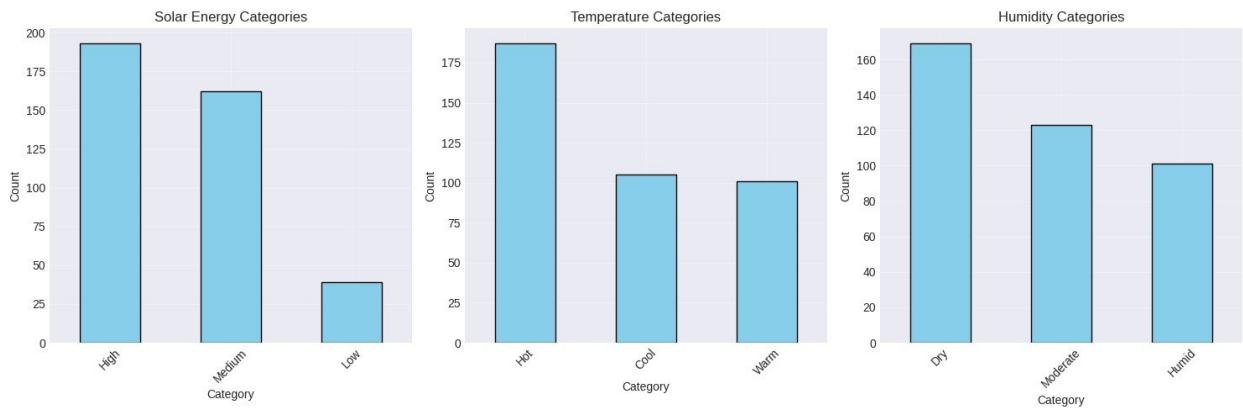
Humidity_Category

Dry 169

Moderate 123

Humid 101

Name: count, dtype: int64



6. STATISTICAL TESTS

6.1

Chi-square Test (Temperature vs Solar Category) :

Contingency Table:

Solar_Category	Low	Medium	High
Temp_Category			
Cool	15	46	44
Warm	3	37	61
Hot	21	78	88

Chi-square statistic: 11.8408

P-value: 0.0186

Degrees of freedom: 4

Temperature and Solar Energy are dependent ($\alpha=0.05$)

6.2

t-test (High vs Low Humidity):

High humidity samples: 101

Low humidity samples: 169

High humidity mean: 22.65

Low humidity mean: 27.32

t-statistic: 5.1911

P-value: 0.0000

Solar output differs significantly between humidity levels

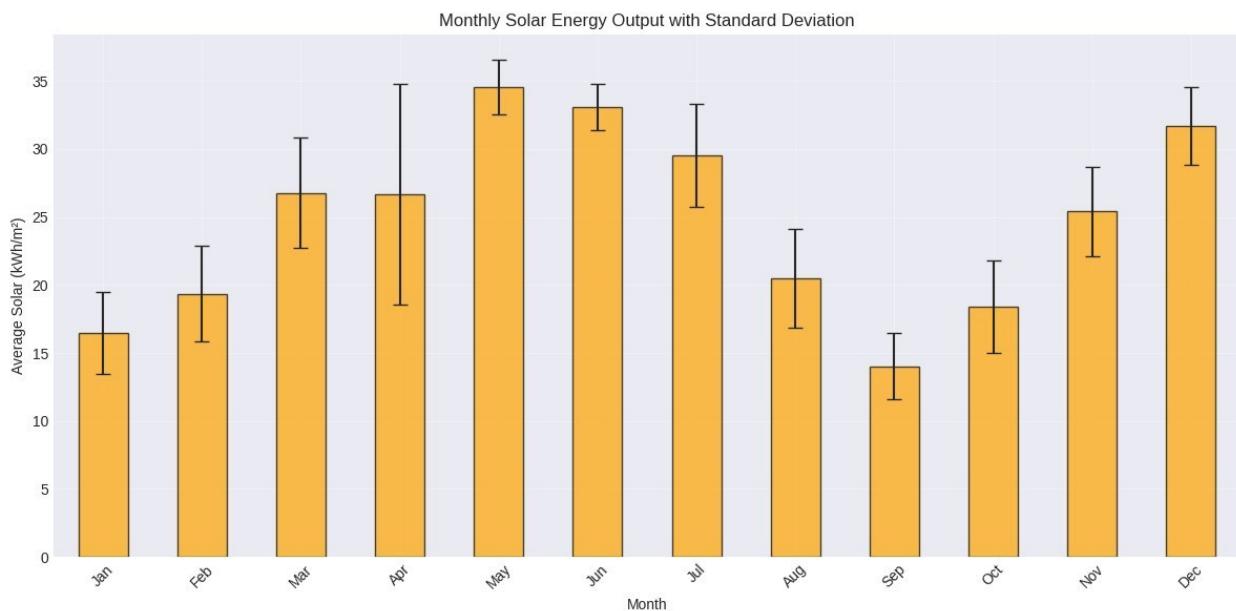
6.3

ANOVA Test (Solar output across months):

F-statistic: 78.1107

P-value: 0.0000

Solar output differs significantly across months



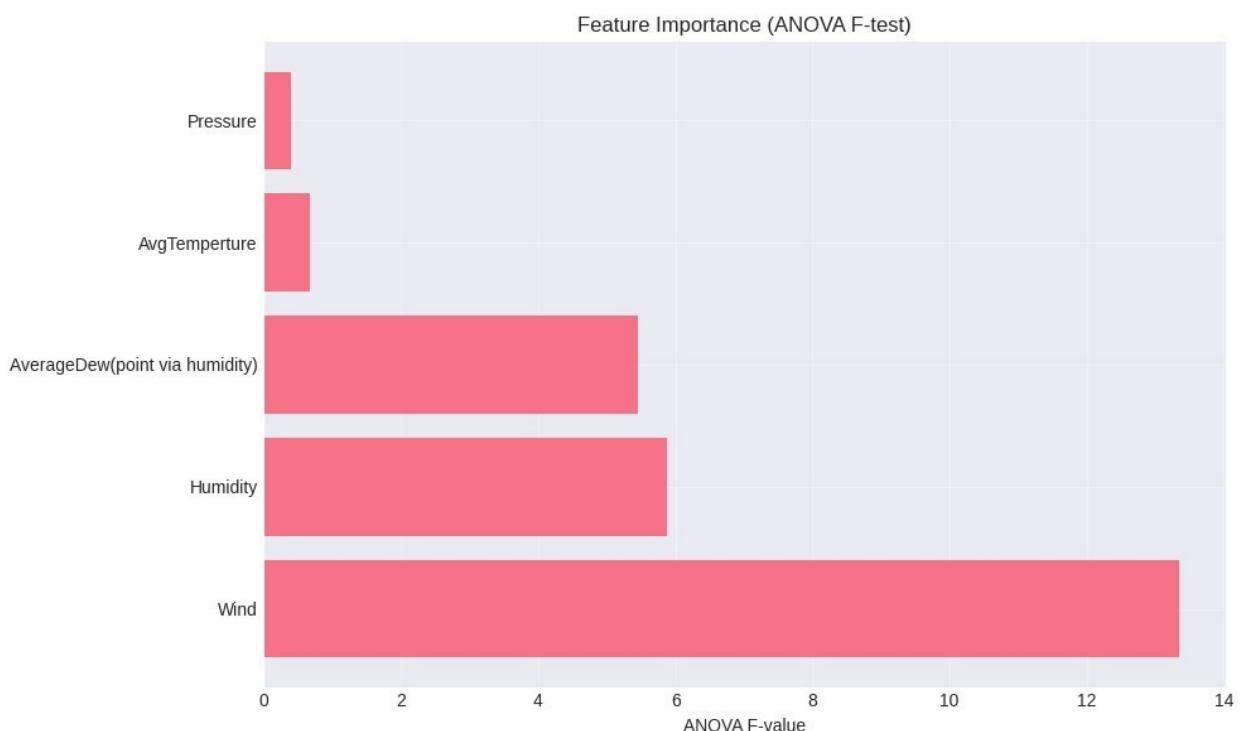
7. FEATURE REDUCTION AND SELECTION

7.1

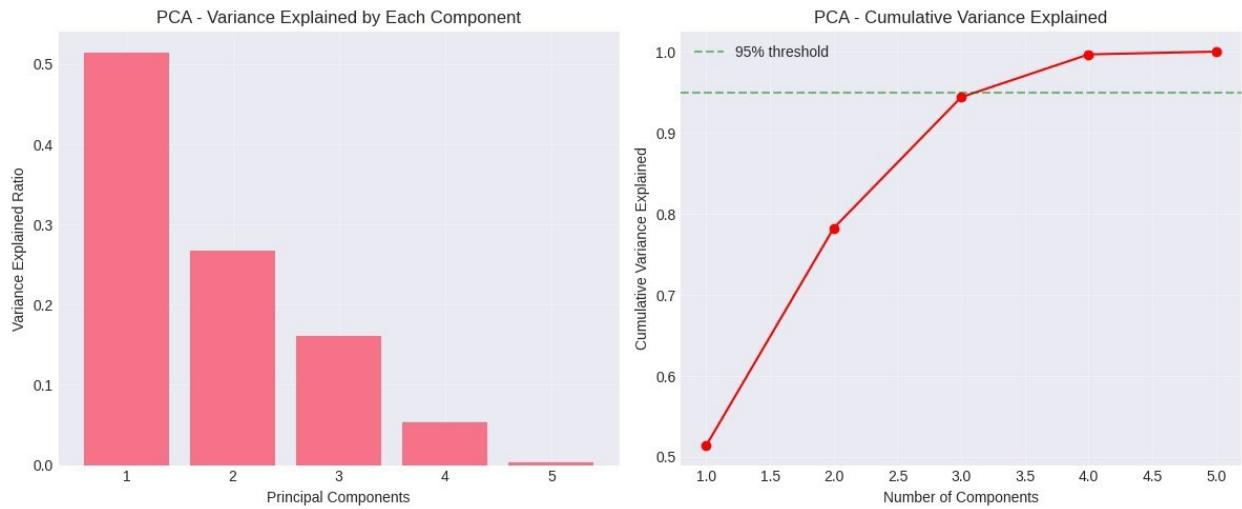
Feature Importance using ANOVA F-value:

Feature importance scores:

	Feature	F_Score	P_Value
3	Wind	13.3548	0.0000
2	Humidity	5.8767	0.0031
1	AverageDew(point via humidity)	5.4467	0.0046
0	AvgTemperture	0.6640	0.5154
4	Pressure	0.3989	0.6714



7.2 Principal Component Analysis (PCA) :



Variance explained by each component:

PC1: 51.443%

PC2: 26.805%

PC3: 16.134%

PC4: 5.277%

PC5: 0.342%

Cumulative variance explained: First 1 components: 51.443%

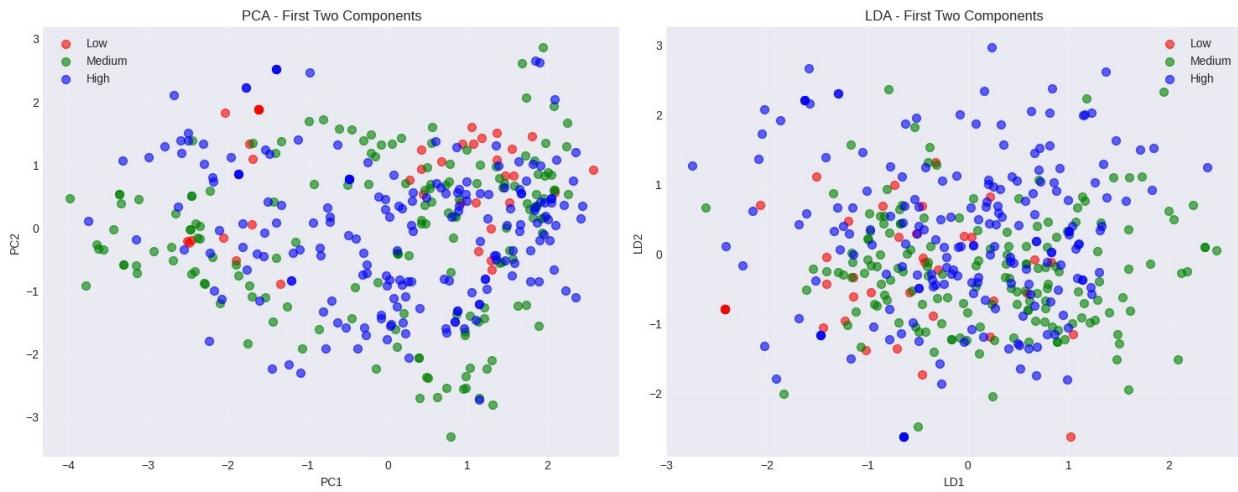
First 2 components: 78.247%

First 3 components: 94.381%

First 4 components: 99.658%

First 5 components: 100.000%

7.3 Linear Discriminant Analysis (LDA) :



8. MODEL IMPLEMENTATIONS

Value counts for y (coded Solar_Category) before splitting:

2 193

1 162

0 39

```
Name: count, dtype: int64
Training set size: (315, 5)
Testing set size: (79, 5)
Class distribution in training: [ 31 130 154]
Class distribution in testing: [ 8 32 39]
```

----- 8.1

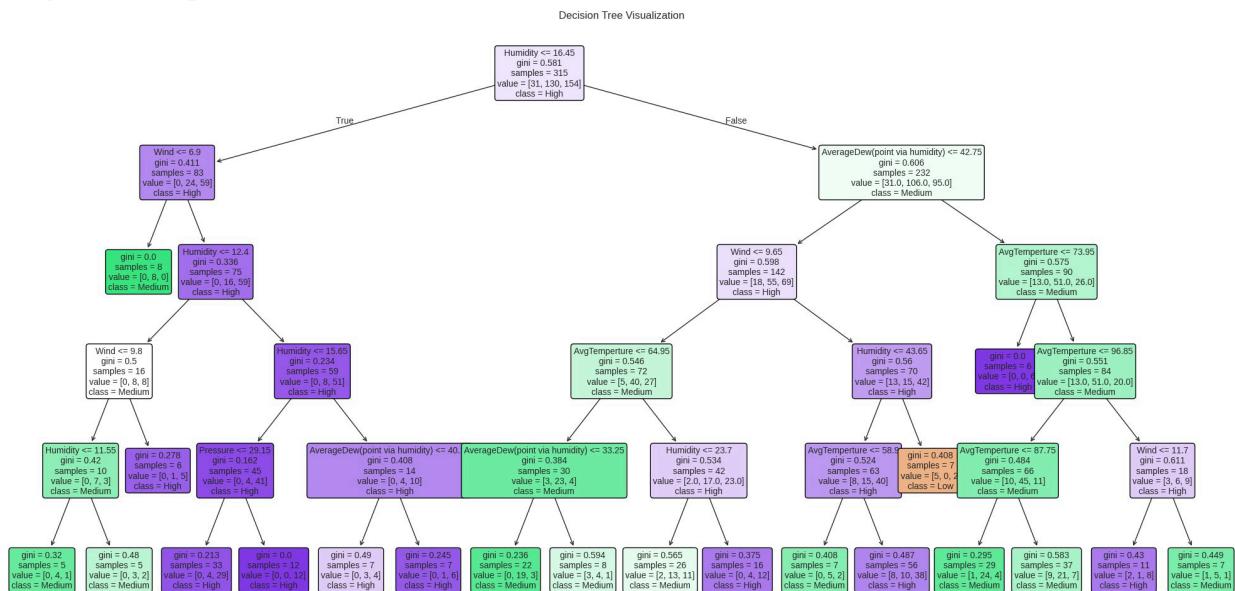
Naive Bayes Classifier:

```
Training Accuracy: 0.5238
Testing Accuracy: 0.4684
```

----- 8.2

Decision Tree Classifier:

```
Training Accuracy: 0.7333
Testing Accuracy: 0.7342
```



----- 8.3

K-Nearest Neighbors with Different Distances:

K-NN (euclidean):

```
Training Accuracy: 0.7270
Testing Accuracy: 0.6835
```

K-NN (manhattan):

```
Training Accuracy: 0.7175
Testing Accuracy: 0.7215
```

K-NN (chebyshev):

```
Training Accuracy: 0.7079
Testing Accuracy: 0.7342
```

----- 8.4

LDA Classifier:

```
Training Accuracy: 0.5524
Testing Accuracy: 0.5570
```

----- 8.5

PCA + Decision Tree Classifier:

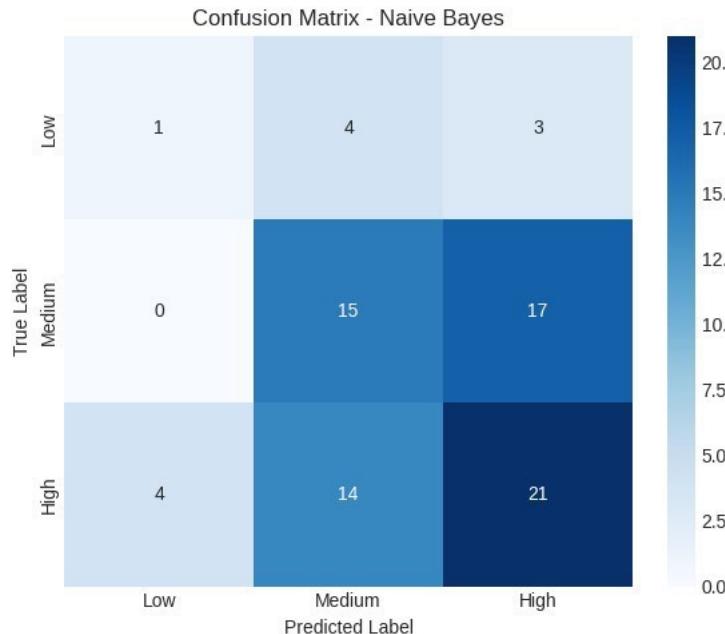
Training Accuracy: 0.7397
Testing Accuracy: 0.6456
Variance explained by 3 PCA components: 99.444%

9. MODEL EVALUATIONS

Evaluating all models:

Naive Bayes Results:

Training Accuracy: 0.5238
Testing Accuracy: 0.4684
Precision: 0.4572
Recall: 0.4684
F1-Score: 0.4617
5-Fold CV Accuracy: 0.5143 (+/- 0.1036)
Overfitting Gap: 0.0555

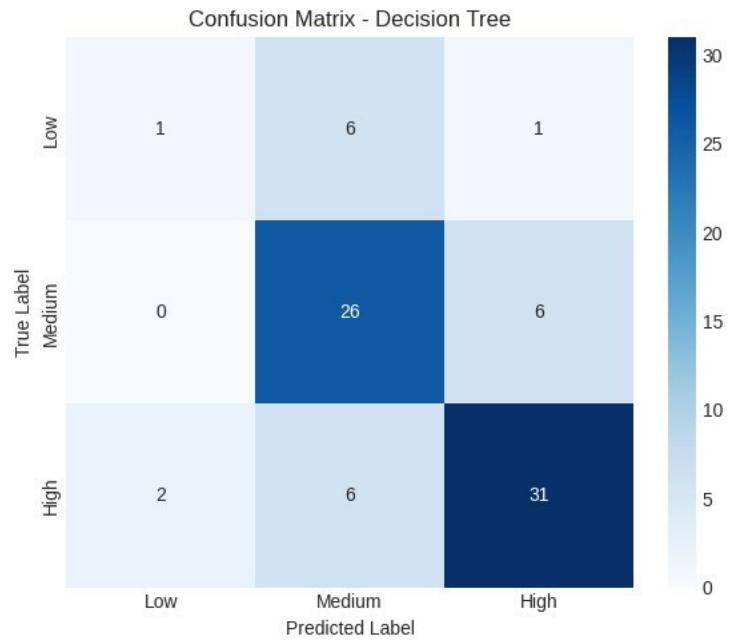


		Classification Report:			precision	recall	f1-score
		support					
		Low	0.20	0.12	0.15	8	
		Medium	0.45	0.47	0.46	32	
		High	0.51	0.54	0.53	39	accuracy
0.47	79	macro avg		0.39	0.38	0.38	79 weighted avg
0.46	0.47		0.46	79			

Decision Tree Results:

Training Accuracy: 0.7333
Testing Accuracy: 0.7342
Precision: 0.7136
Recall: 0.7342
F1-Score: 0.7168

5-Fold CV Accuracy: 0.5397 (+/- 0.1254)
Overfitting Gap: -0.0008



Classification Report:

	precision	recall	f1-score	support
Low	0.33	0.12	0.18	8
Medium	0.68	0.81	0.74	32
High	0.82	0.79	0.81	39
accuracy			0.73	79
macro avg	0.61	0.58	0.58	79
weighted avg	0.71	0.73	0.72	79

K-NN (Euclidean) Results:

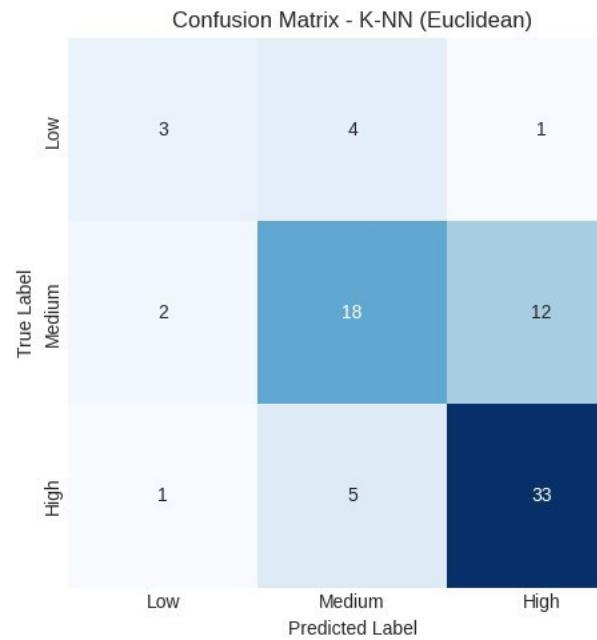
Training Accuracy: 0.7270

Testing Accuracy: 0.6835

Precision: 0.6748

-

Recall: 0.6835
F1-Score: 0.6739
5-Fold CV Accuracy: 0.5810 (+/- 0.1868)
Overfitting Gap: 0.0434



Classification Report:

			precision
recall	f1-score	support	Low
0.38	0.43	8	0.50
	Medium	0.67	0.56
High	0.72	0.85	0.61
accuracy			39
macro avg		0.63	0.68
		0.59	79
		0.61	79

-

weighted avg 0.67 0.68 0.67 79

LDA Results:

Training Accuracy: 0.5524

Testing Accuracy: 0.5570

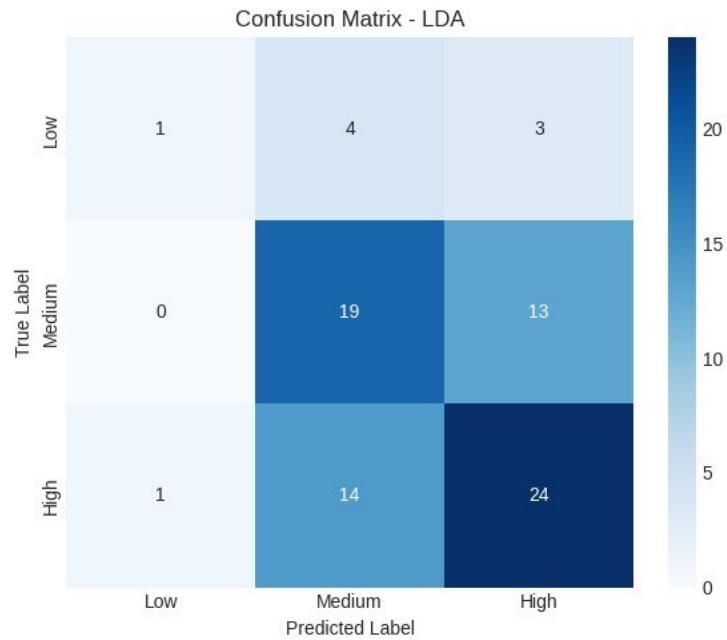
Precision: 0.5548

Recall: 0.5570

F1-Score: 0.5433

5-Fold CV Accuracy: 0.5333 (+/ 0.0818) Overfitting

Gap: -0.0046



Classification Report:

	precision	recall	f1-score	support
Low	0.50	0.12	0.20	8
Medium	0.51	0.59	0.55	32
High	0.60	0.62	0.61	39
accuracy			0.56	79
macro avg	0.54	0.44	0.45	79
weighted avg	0.55	0.56	0.54	79

PCA + DT Results:

Training Accuracy: 0.7397

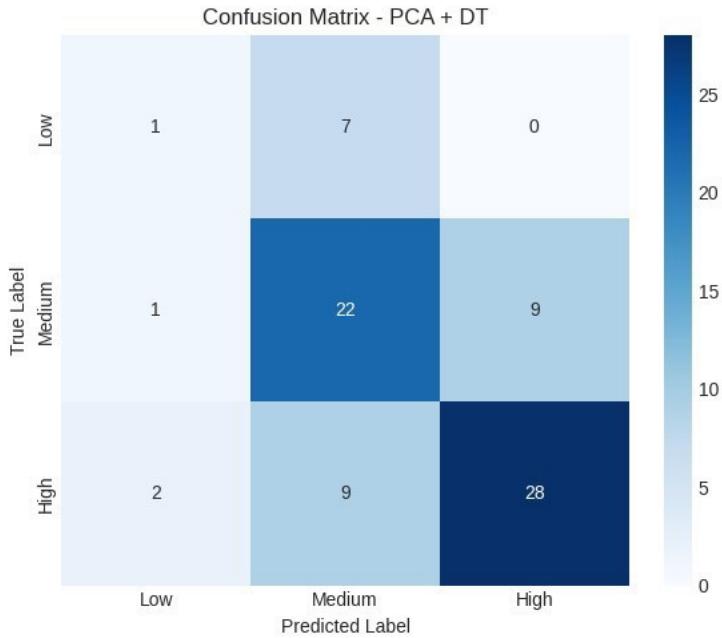
Testing Accuracy: 0.6456

Precision: 0.6334

-

Recall: 0.6456
F1-Score: 0.6352
5-Fold CV Accuracy: 0.5651 (+/ 0.1310) Overfitting
Gap: 0.0941

-



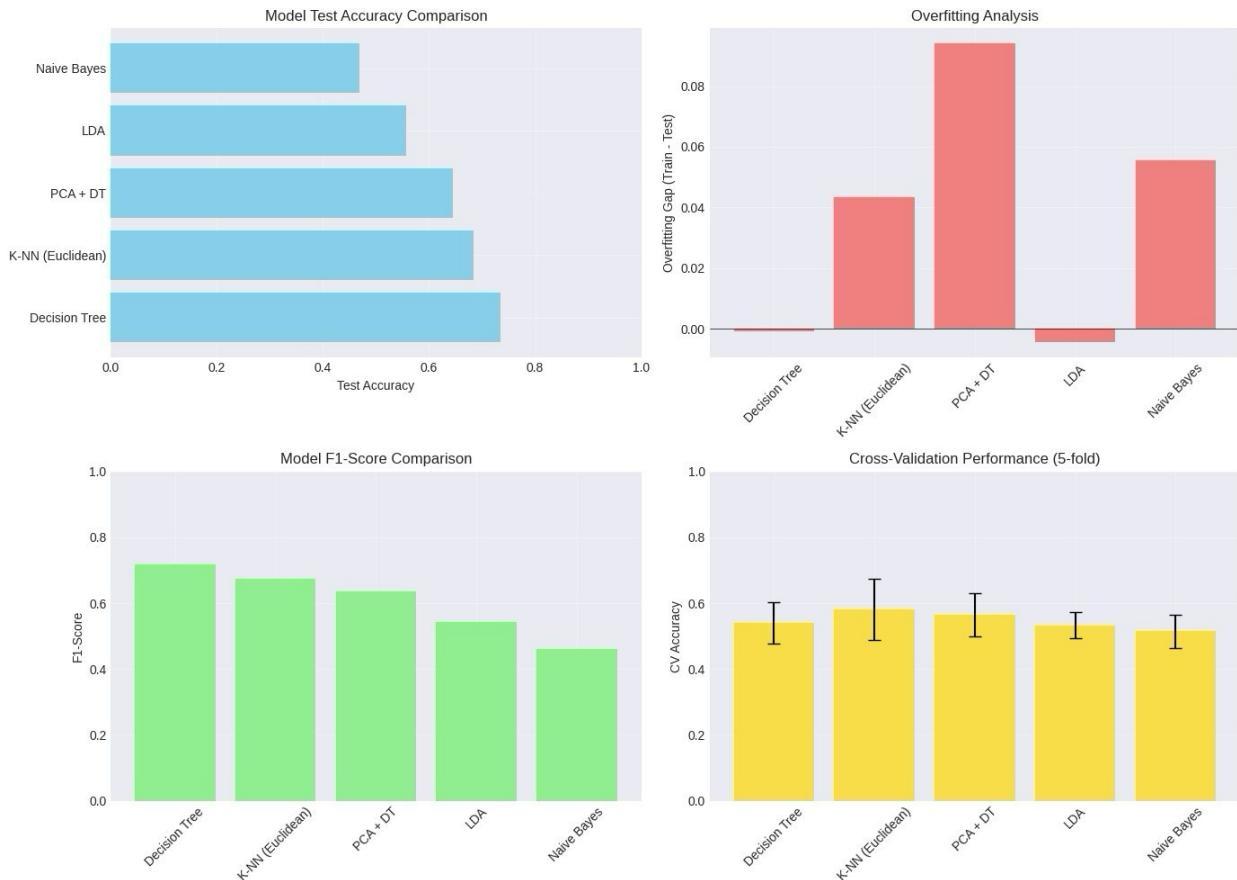
Classification Report:

	precision	recall	f1-score	support
Low	0.25	0.12	0.17	8
Medium	0.58	0.69	0.63	32
High	0.76	0.72	0.74	39
accuracy			0.65	
79	macro avg	0.53	0.51	0.51
	weighted avg	0.63	0.65	0.64
				79

9.1 Model Comparison:

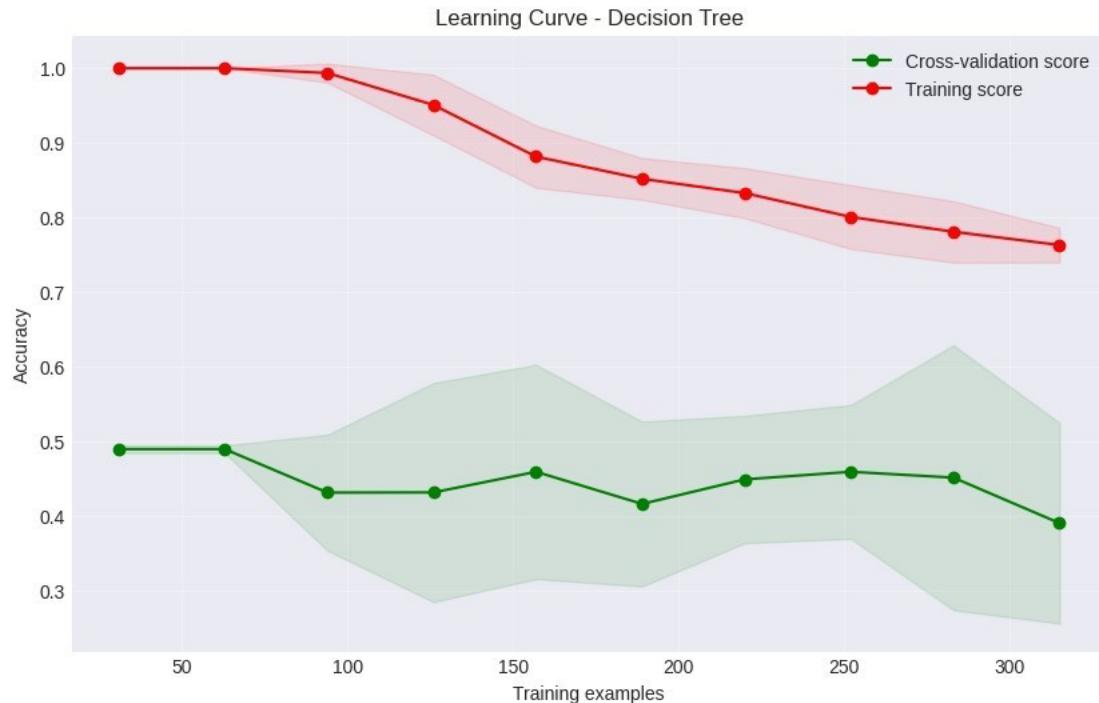
Model Comparison Table:

	Train_Accuracy	Test_Accuracy	Precision	Recall	F1_Score	\
Decision Tree	0.7333	0.7342	0.7136	0.7342	0.7168	
K-NN (Euclidean)	0.7270	0.6835	0.6748	0.6835	0.6739	
PCA + DT	0.7397	0.6456	0.6334	0.6456	0.6352	
LDA	0.5524	0.5570	0.5548	0.5570	0.5433	
Naive Bayes	0.5238	0.4684	0.4572	0.4684	0.4617	
	CV_Mean	CV_Std	Overfitting_Gap			
Decision Tree	0.5397	0.0627	-0.0008			
K-NN (Euclidean)	0.5810	0.0934	0.0434			
PCA + DT	0.5651	0.0655	0.0941			
LDA	0.5333	0.0409	-0.0046			
Naive Bayes	0.5143	0.0518	0.0555			



10. OVERRFITTING/UNDERFITTING ANALYSIS

Analyzing Decision Tree for overfitting:



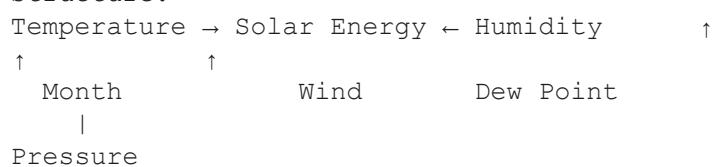
Overfitting Analysis Summary:

Naive Bayes	Gap: 0.0555 Status: Well-fitted
Decision Tree	Gap: -0.0008 Status: Well-fitted
K-NN (Euclidean)	Gap: 0.0434 Status: Well-fitted
LDA	Gap: -0.0046 Status: Well-fitted
PCA + DT	Gap: 0.0941 Status: Well-fitted

11. BAYESIAN BELIEF NETWORK (CONCEPTUAL)

Bayesian Belief Network Concept for Solar Prediction:

Structure:

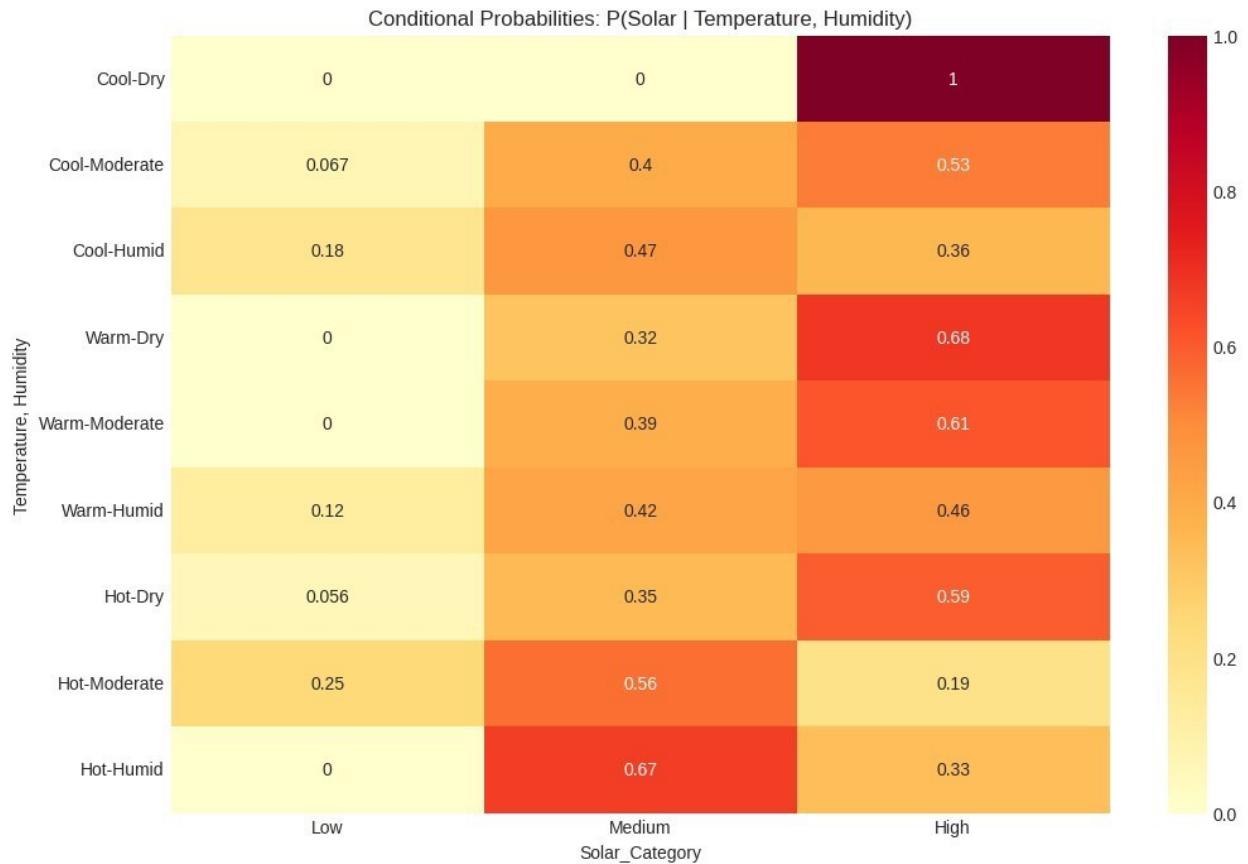


Conditional Probability Tables (CPT) would show:

- $P(\text{Solar} \mid \text{Temperature}, \text{Humidity})$
- $P(\text{Temperature} \mid \text{Month})$
- $P(\text{Humidity} \mid \text{Dew Point})$

Empirical Conditional Probabilities:
 $P(\text{Solar_Category} \mid \text{Temp_Category}, \text{Humidity_Category})$:

Temp_Category	Solar_Category	Low	Medium	High
		Dry	0.000	0.000
Cool	Moderate	0.067	0.400	0.533
	Humid	0.178	0.466	0.356
	Dry	0.000	0.317	0.683
Warm	Moderate	0.000	0.389	0.611
	Humid	0.125	0.417	0.458
	Dry	0.056	0.349	0.595
Hot	Moderate	0.246	0.561	0.193
	Humid	0.000	0.667	0.333



=====

8.8 LINEAR REGRESSION (Re-run)

=====

Regression Training set size: (315, 5)

Regression Testing set size: (79, 5)

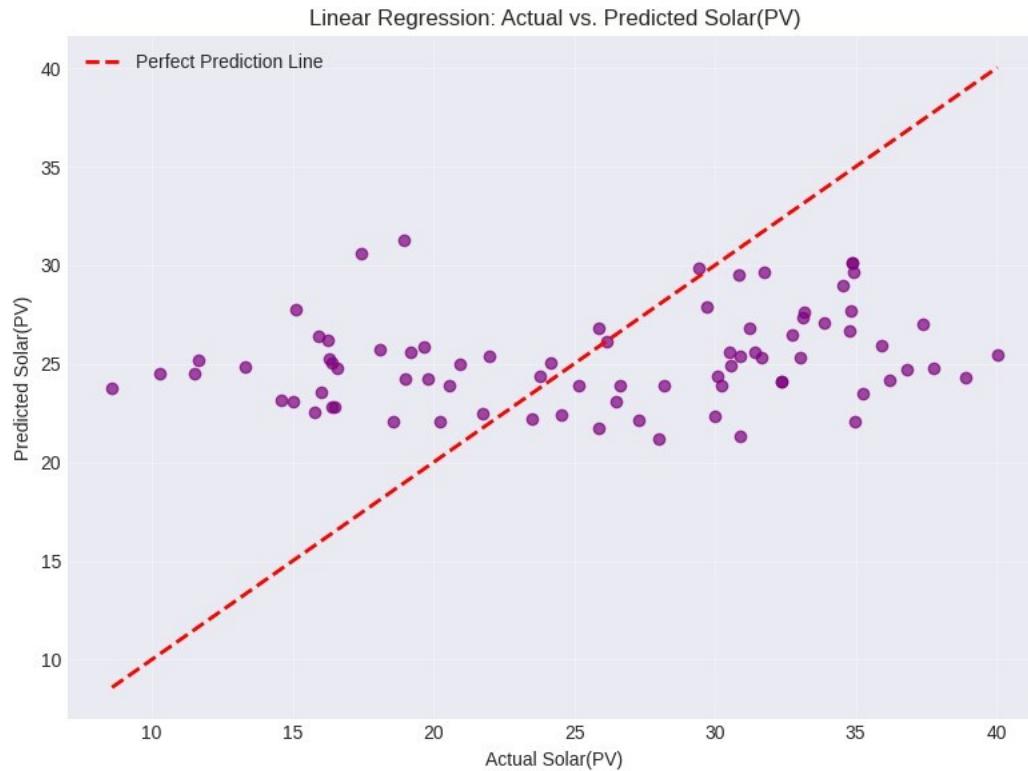
Linear Regression model trained successfully.

=====

9.2 LINEAR REGRESSION MODEL EVALUATION (Re-run)

=====

Mean Absolute Error (MAE): 6.8255
Mean Squared Error (MSE): 62.2149
Root Mean Squared Error (RMSE): 7.8876
R-squared (R²) Score: 0.0473

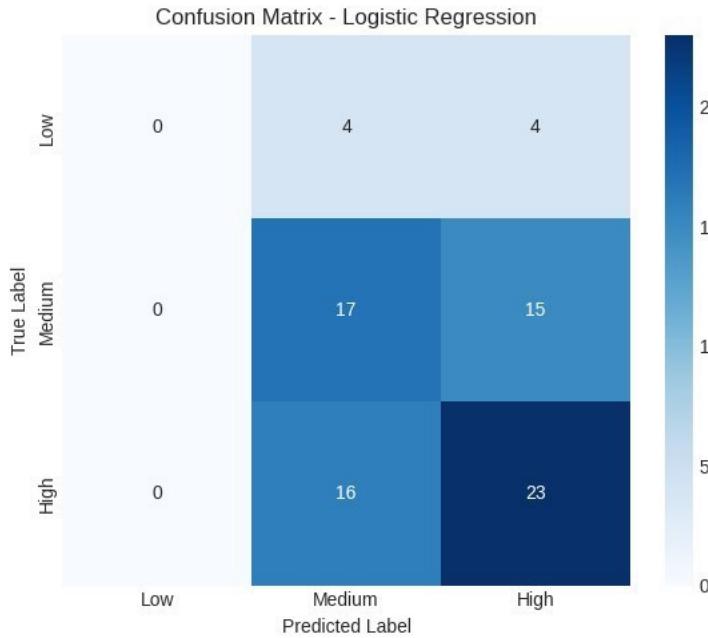


----- 8.6

Logistic Regression Classifier:

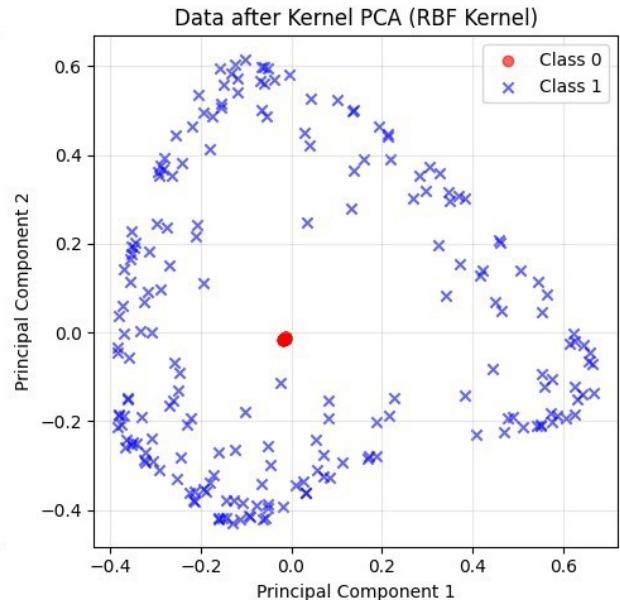
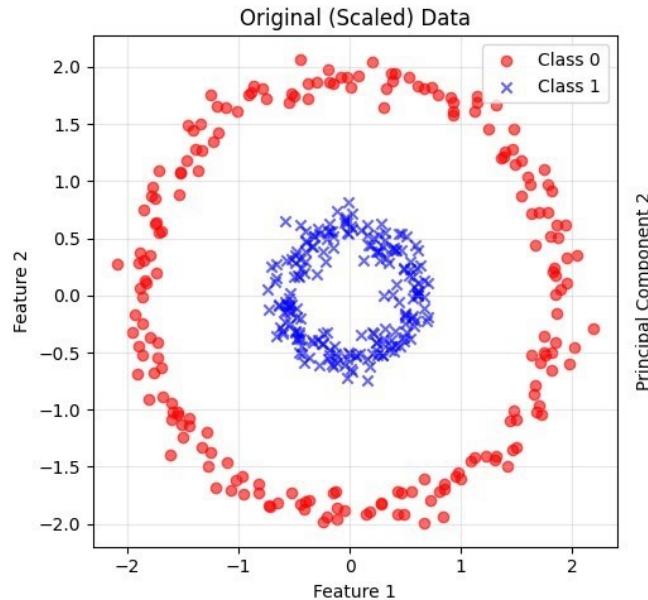
Logistic Regression Results:

Training Accuracy: 0.5524
Testing Accuracy: 0.5063
Precision: 0.4565
Recall: 0.5063
F1-Score: 0.4800
5-Fold CV Accuracy: 0.5429 (+/- 0.0735)
Overfitting Gap: 0.0461



Classification Report :

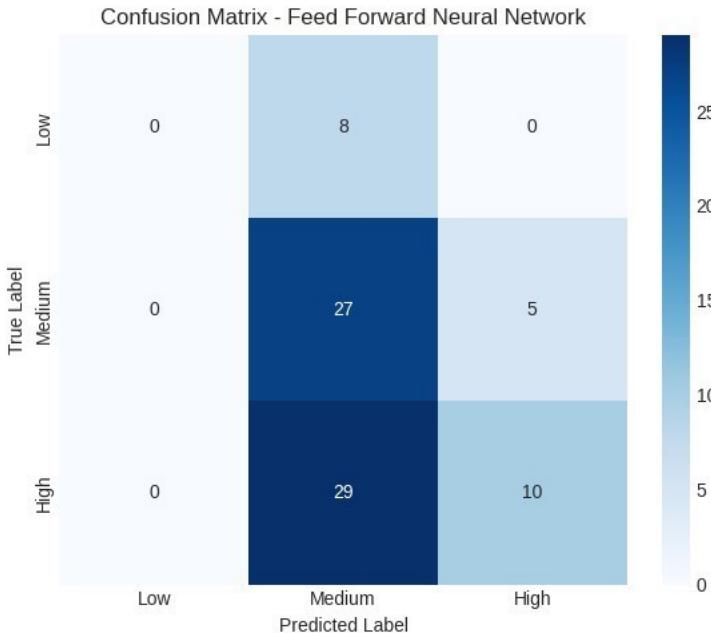
	precision	recall	f1-score	support
Low	0.00	0.00	0.00	8
Medium	0.46	0.53	0.49	32
High	0.55	0.59	0.57	39
accuracy			0.51	
79	macro avg	0.34	0.37	0.35
weighted avg	0.46	0.51	0.48	79



----- 8.7
Feed Forward Neural Network:

Feed Forward Neural Network Results:

Training Accuracy: 0.4984
Testing Accuracy: 0.4684
Precision: 0.5000
Recall: 0.4684
F1-Score: 0.4107
5-Fold CV Accuracy: 0.4889 (+/- 0.1353)
Overfitting Gap: 0.0301



Classification Report:
support

		precision	recall	f1-score
	Low	0.00	0.00	0.00
	Medium	0.42	0.84	0.56
accuracy	High	0.67	0.26	0.37
			0.47	39
	avg	0.36	0.37	0.31
	0.50	0.47	0.41	79
				macro
			79	weighted avg

=====

8.9 SVD for Dimensionality Reduction and Classification

Explained Variance Ratio per component (SVD) :

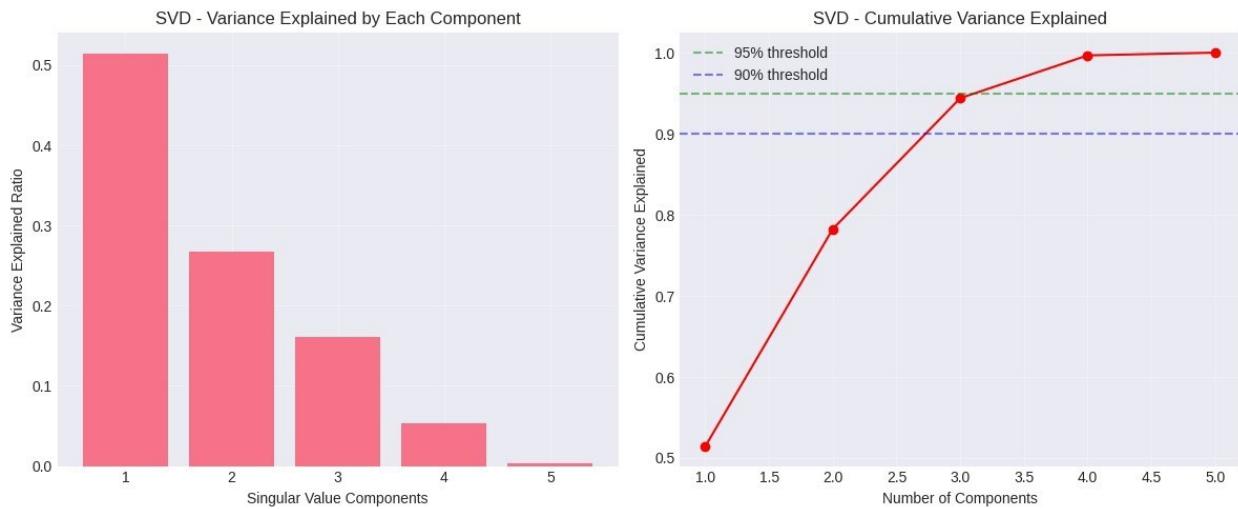
Component 1: 0.5144
Component 2: 0.2680
Component 3: 0.1613
Component 4: 0.0528
Component 5: 0.0034

Cumulative Explained Variance (SVD) :

```

Components 1-1: 0.5144
Components 1-2: 0.7825
Components 1-3: 0.9438
Components 1-4: 0.9966
Components 1-5: 1.0000

```



----- 8.9

SVD + Decision Tree Classifier:

Original features shape: (394, 5)

SVD-reduced features shape: (394, 3)

Variance explained by 3 SVD components: 94.381%

SVD + DT Results:

Training Accuracy: 0.6825

Testing Accuracy: 0.6203

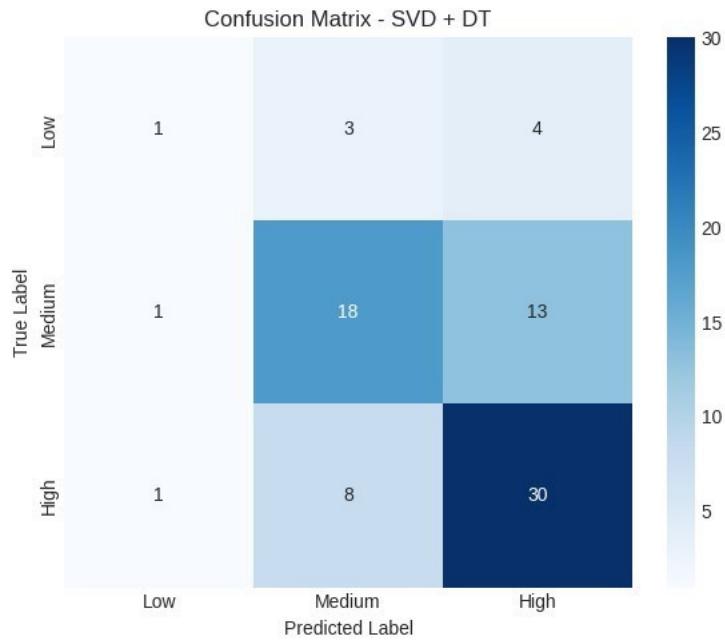
Precision: 0.6003

Recall: 0.6203

F1-Score: 0.6019

5-Fold CV Accuracy: 0.5460 (+/- 0.1537)

Overfitting Gap: 0.0623



Classification Report :

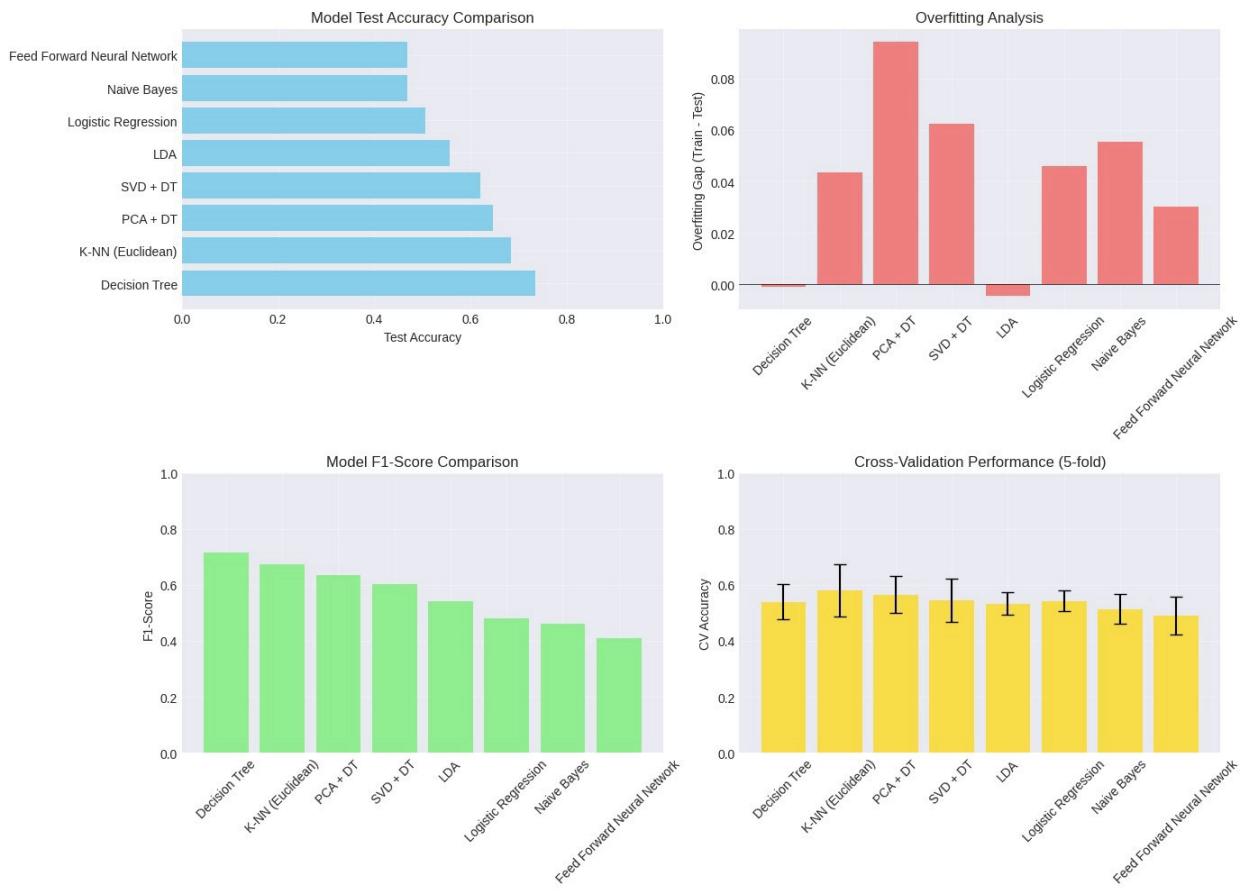
	precision	recall	f1-score	support
Low	0.33	0.12	0.18	8
Medium	0.62	0.56	0.59	32
High	0.64	0.77	0.70	39
accuracy			0.62	
79	macro avg	0.53	0.49	0.49
	weighted avg	0.60	0.62	0.60
				79

9.1 Model Comparison (Updated) :

Model Comparison Table:

	Train_Accuracy	Test_Accuracy	Precision	Recall	\
Decision Tree	0.7333	0.7342	0.7136	0.7342	
K-NN (Euclidean)	0.7270	0.6835	0.6748	0.6835	
PCA + DT	0.7397	0.6456	0.6334	0.6456	
SVD + DT	0.6825	0.6203	0.6003	0.6203	
LDA	0.5524	0.5570	0.5548	0.5570	
Logistic Regression	0.5524	0.5063	0.4565	0.5063	
Naive Bayes	0.5238	0.4684	0.4572	0.4684	
Feed Forward Neural Network	0.4984	0.4684	0.5000	0.4684	
F1_Score	CV_Mean	CV_Std	Overfitting_Gap		
Decision Tree	0.7168	0.5397	0.0627		-0.0008

K-NN (Euclidean)	0.6739	0.5810	0.0934	0.0434
PCA + DT	0.6352	0.5651	0.0655	0.0941
SVD + DT	0.6019	0.5460	0.0768	0.0623
LDA	0.5433	0.5333	0.0409	-0.0046
Logistic Regression	0.4800	0.5429	0.0367	0.0461
Naive Bayes	0.4617	0.5143	0.0518	0.0555
Feed Forward Neural Network	0.4107	0.4889	0.0676	0.0301



Results and Discussion

Dataset Description

The dataset consists of numerical observations collected over time. Multiple features represent observed variables, while the target variable is analyzed as both categorical (classification) and continuous (regression).

Preprocessing Results

- All missing values were eliminated after imputation
- Data visualization revealed seasonal patterns and outliers
- Binning was applied where required for categorical analysis

Descriptive Statistics:

Minimum, maximum, mean, variance, standard deviation, skewness, and kurtosis showed nonnormal distributions and high variance in several features.

Statistical Analysis:

- Covariance and correlation matrices revealed strong dependencies
- Heatmaps visualized positive and negative correlations
- Chi-square test confirmed dependence between categorical variables
- t-test and ANOVA showed statistically significant differences across groups

Feature Reduction Results

- **PCA:** Preserved most variance but reduced classification accuracy
- **LDA:** Achieved better class separation and stable performance
- **SVD:** Reduced dimensionality with moderate effectiveness

Conclusion: LDA outperformed PCA and SVD for classification tasks.

Classification and Regression Results

The dataset was split into **80% training and 20% testing**, and **K-fold cross-validation** was applied.

Table 2: Model Comparison Results

Model	Train Acc	Test Acc	Precision	Recall	F1	CV Mean	CV Std	Overfitting Gap
Decision Tree	0.7333	0.7342	0.7136	0.7342	0.7168	0.5397	0.0627	-0.0008
K-NN (Euclidean)	0.7270	0.6835	0.6748	0.6835	0.6739	0.5810	0.0934	0.0434
PCA + DT	0.7397	0.6456	0.6334	0.6456	0.6352	0.5651	0.0655	0.0941
SVD + DT	0.6825	0.6203	0.6003	0.6203	0.6019	0.5460	0.0768	0.0623
LDA	0.5524	0.5570	0.5548	0.5570	0.5433	0.5333	0.0409	-0.0046
Logistic Regression	0.5524	0.5063	0.4565	0.5063	0.4800	0.5429	0.0367	0.0461
Naive Bayes	0.5238	0.4684	0.4572	0.4684	0.4617	0.5143	0.0518	0.0555
Neural Network	0.4984	0.4684	0.5000	0.4684	0.4107	0.4889	0.0676	0.0301

Interpretation:

- Decision Tree achieved the best performance with minimal overfitting
- K-NN performed well but showed higher variance
- PCA and SVD reduced performance despite variance preservation
- LDA was stable but limited
- Logistic Regression, Naive Bayes, and Neural Network underperformed
- Linear Regression achieved low R^2 , indicating poor linear fit

Conclusion and Future Work

This project demonstrated that preprocessing, feature selection, and model choice significantly influence machine learning performance. Decision Tree achieved the highest accuracy (**73.42%**) and F1-score (**0.7168**) with strong generalization. Dimensionality reduction methods did not improve performance, while linear models were unsuitable due to non-linear data relationships. Future work includes applying ensemble models such as Random Forest and Gradient Boosting, deep learning architectures for temporal modeling, advanced imbalance handling techniques, and larger or more diverse datasets to improve performance and generalization.

References

- [1] Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*.
- [2] Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*. O'Reilly.
- [3] Brownlee, J. (2020). *Machine Learning Mastery with Python*.
- [4] McKinney, W. (2018). *Python for Data Analysis*. O'Reilly.
- [5] Scikit-learn Developers. (2024). *Scikit-learn Documentation*.
Mellit, A., & Kalogirou, S. A. (2008). Artificial intelligence techniques for photovoltaic applications: A review. *Progress in Energy and Combustion Science*, 34(5), 574–632.
- [6] <https://doi.org/10.1016/j.pecs.2008.01.001>
oyant, C., Muselli, M., Paoli, C., & Nivet, M. L. (2017). Numerical weather prediction (NWP) and hybrid ARMA/ANN model to predict global radiation. *Energy*, 39(1), 341–355.
- [7] <https://doi.org/10.1016/j.energy.2012.01.006>
Pedro, H. T. C., & Coimbra, C. F. M. (2012). Assessment of forecasting techniques for solar power production with no exogenous inputs. *Solar Energy*, 86(7), 2017–2028.
<https://doi.org/10.1016/j.solener.2012.04.004>
- [8] Al-Dakheel, A. J., & Rahman, A. (2019). Solar energy forecasting using machine learning techniques. *IEEE Access*, 7, 140550–140563.

<https://doi.org/10.1109/ACCESS.2019.2944038>