UTM
UNIVERSITI TEKNOLOGI MALAYSIA

PSM 2 PRESENTATION

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

PRESENTED BY : YUSRA NADATUL ALYEEA BINTI YUSRAMIZAL
SUPERVISED BY: DR. ZURAINI BINTI ALI SHAH

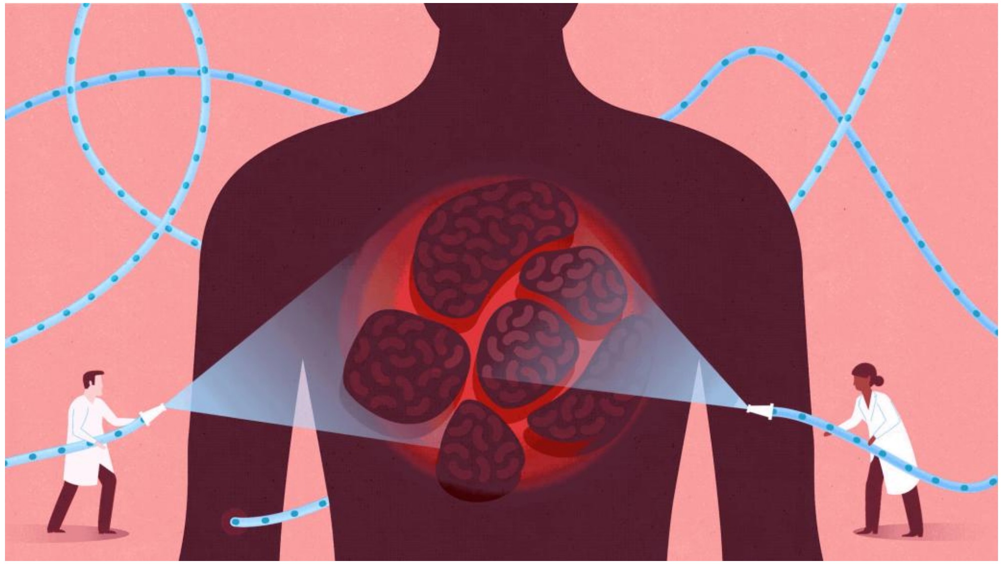
UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 1
INTRODUCTION

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

Problem Background

Cancer remains a leading cause of death worldwide, with increasing incidence and mortality rates. Despite advances in genomic technologies that generate massive gene expression data, challenges like high dimensionality, class imbalance, and noise make accurate classification difficult. Reliable classification is crucial for early diagnosis, personalized therapy, and treatment planning.



Problem Statement

The volume and complexity of gene expression datasets, which are often characterized by **high-dimensional spaces** and **large feature sets**, pose significant challenges for accurate classification.

Research Goal

To **improve the accuracy** of cancer gene expression data classification using **Particle Swarm Optimization (PSO)** techniques.

Research Objectives

01

To select the informative gene using Particle Swarm Optimization (PSO) on cancer gene expression.

02

To assess the impact of PSO feature selection on machine learning classification model performance.

03

To compare the performance of PSO-enhanced classification algorithms with traditional methods.

CHAPTER 2
LITERATURE REVIEW

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

Literature Review on Cancer Classification Performance

- Emphasizes comparison of classifier performance with and without PSO
- **Studies reviewed:** Kazerani (2024), Alrefai & Ibrahim (2022)
- **Datasets involved:** Clinical Datasets (WDBC, Coimbra) and Microarray Datasets (Colon, Breast)
- **Performance metrics:** Accuracy, Sensitivity, Precision

Dataset Types in Reviewed Studies

Dataset	Type	Source
Breast (WDBC)	Clinical	Kazerani (2024)
Breast (Coimbra)		
Breast	Microarray	Alrefai and Ibrahim (2022)
Colon		

**For this presentation, I use Coimbra dataset to show classifier comparisons.*

Classifier Performance (Coimbra Dataset with PSO)

Classifier	Accuracy (%)	Sensitivity (%)	Precision (%)
SVM	**91**	**100**	**85**
ANN	**90**	**92**	**89**
AdaBoost	**88**	**78**	**100**
Decision Tree	87	76	100
KNN	87	87	89
Random Forest	87	95	84
Linear Regression	74	78	76
Logistic Regression	73	62	85
Naïve Bayes	72	59	86

Key Finding from Literature

1 **SVM**: Highest sensitivity (100%), strong accuracy (91%)

2 **ANN**: High accuracy (90%) and good precision (89%)

3 **AdaBoost**: Perfect precision (100%), strong accuracy (88%)

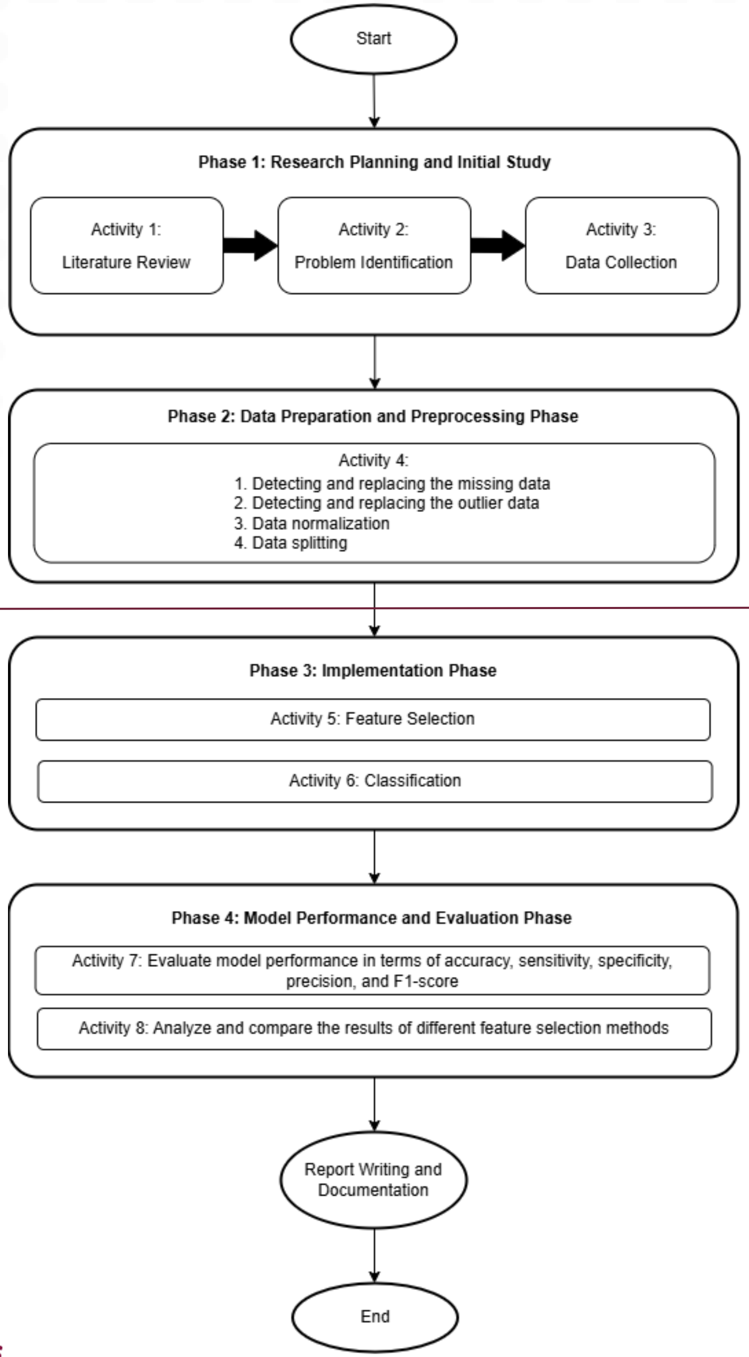
These classifiers were consistently top performers in clinical datasets, making them ideal for this research.

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 3
RESEARCH METHODOLOGY

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

Research Framework



Start

Phase 1: Research Planning and Initial Study

Activity 1: Literature Review

Activity 2: Problem Identification

Activity 3: Data Collection

Phase 2: Data Preparation and Preprocessing Phase

Activity 4:

1. Detecting and replacing the missing data
2. Detecting and replacing the outlier data
3. Data normalization
4. Data splitting

PSM 1

PSM 2

Phase 3: Implementation Phase

Activity 5: Feature Selection

Activity 6: Classification

Phase 4: Model Performance and Evaluation Phase

Activity 7: Evaluate model performance in terms of accuracy, sensitivity, specificity, precision, and F1-score

Activity 8: Analyze and compare the results of different feature selection methods

Report Writing and Documentation

End

UTM UNIVERSITI TEKNOLOGI MALAYSIA

Innovating Solutions

www.utm.my

Dataset

The cancer gene expression dataset was obtained from Curated Microarray Database (CuMiDa) involving two types of cancer, colorectal and lung cancer.

Dataset	No. of Samples	No. of Features	No. of Classes
Colorectal	105	22,278	2
Lung	114	54,676	2

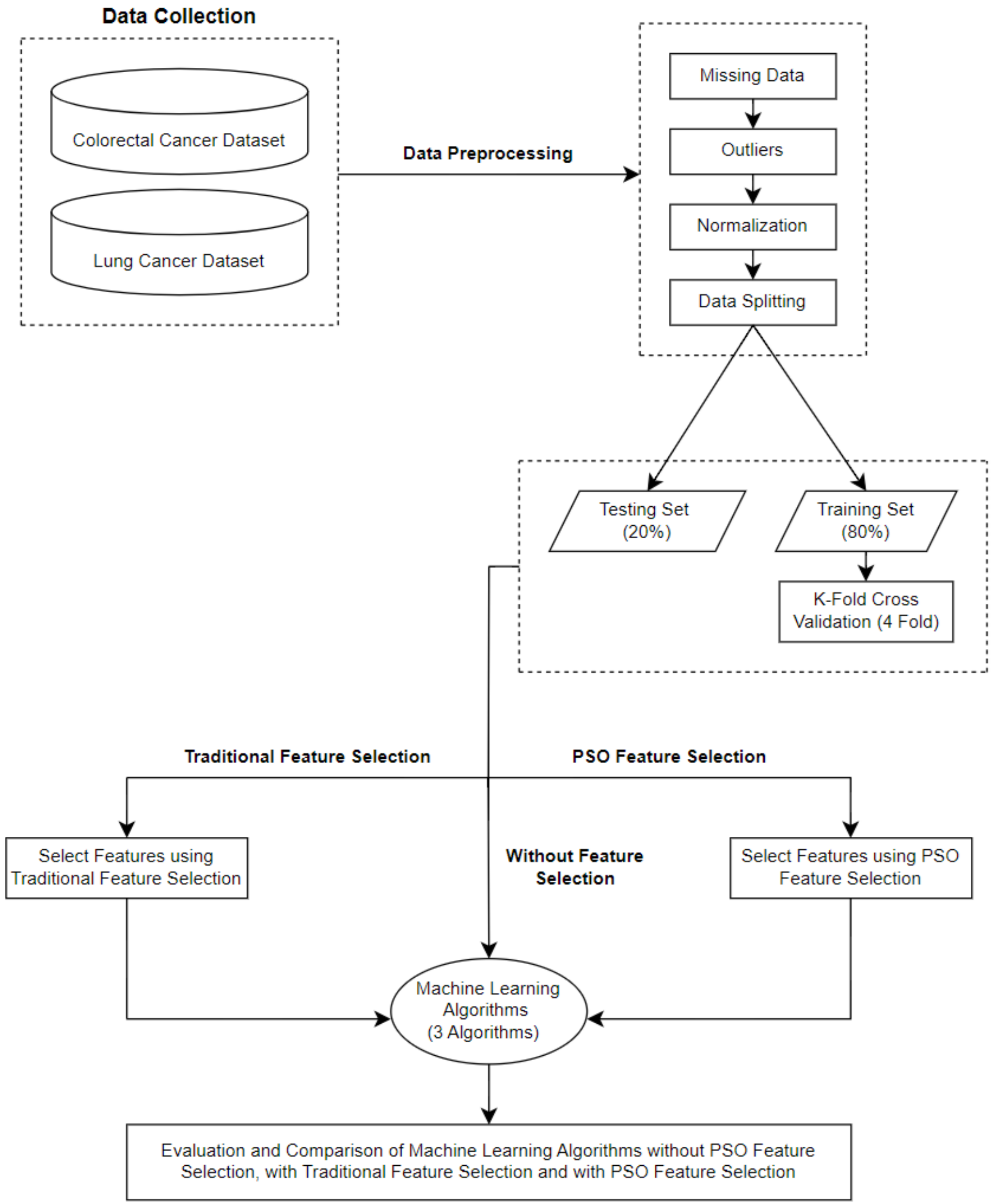
Class Distribution

Dataset	Class Distribution	No. of Samples
Colorectal	normal	53
	tumoral	52
Lung	normal	58
	tumoral	56

CHAPTER 4
PROPOSED WORK

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

Flowchart



Data Collection

Colorectal Cancer Dataset

Lung Cancer Dataset

Data Preprocessing

Missing Data

Outliers

Normalization

Data Splitting

Testing Set (20%)

Training Set (80%)

K-Fold Cross Validation (4 Fold)

Traditional Feature Selection

PSO Feature Selection

Select Features using Traditional Feature Selection

Without Feature Selection

Select Features using PSO Feature Selection

Machine Learning Algorithms (3 Algorithms)

Evaluation and Comparison of Machine Learning Algorithms without PSO Feature Selection, with Traditional Feature Selection and with PSO Feature Selection

Data Preprocessing

01 Detection and Replacing the Missing Data

02 Detection and Replacing the Outlier Data

03 Data Normalization

04 Data Splitting

Detection and Replacing the Missing Data

- Missing data values (NaN) were **replaced using the mean value** of the respective column's feature.

Detection and Replacing the Missing Data

Dataset	Number of Missing Data
Colorectal	0
Lung	0

Detection and Replacing the Outlier Data

- Outliers were detected using the **Z-score algorithm** (values with Z-score > 3 or < -3).
- These outliers were **replaced by the mean** of the corresponding column (mean imputation).

Detection and Replacing the Outlier Data

Dataset	Outliers Before Imputation	Outliers After Imputation
Colorectal	26,350	14,815
Lung	50,135	20,972

Data Normalization

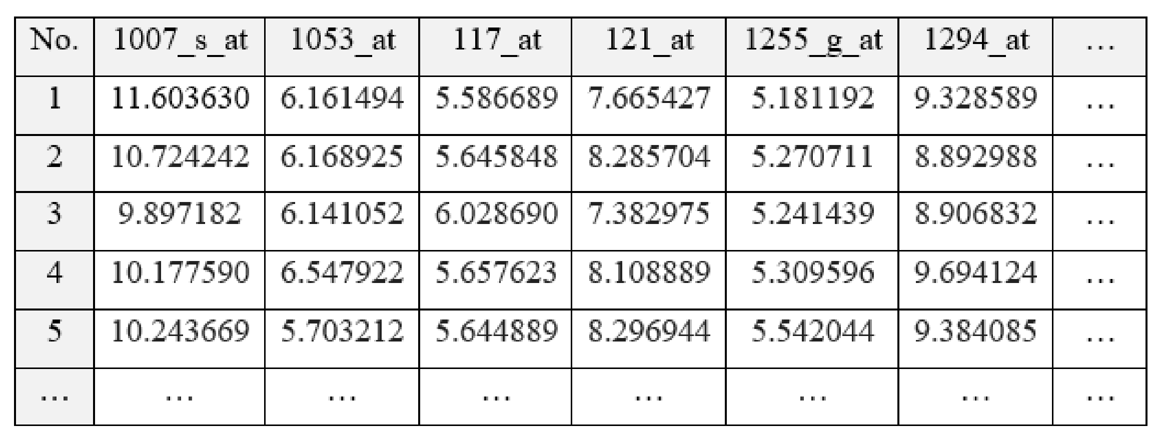
- Data was normalized using **Min-Max normalization**, transforming all observations to a range between 0 and 1.

- Formula:

$$X_N = \frac{X_i - \min(x)}{\max(x) - \min(x)}$$

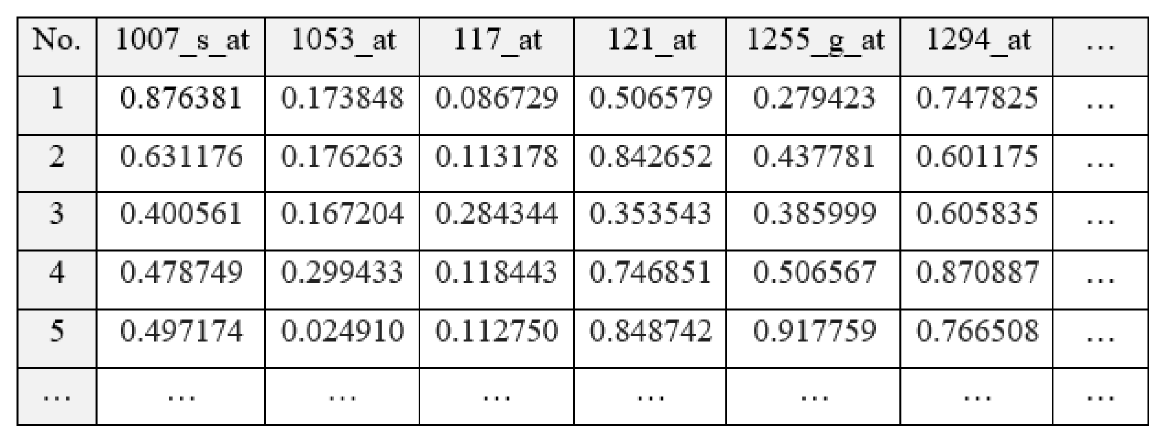
Data Normalization

Colorectal Cancer Dataset Before Normalization



No.	1007_s_at	1053_at	117_at	121_at	1255_g_at	1294_at	…
1	11.603630	6.161494	5.586689	7.665427	5.181192	9.328589	…
2	10.724242	6.168925	5.645848	8.285704	5.270711	8.892988	…
3	9.897182	6.141052	6.028690	7.382975	5.241439	8.906832	…
4	10.177590	6.547922	5.657623	8.108889	5.309596	9.694124	…
5	10.243669	5.703212	5.644889	8.296944	5.542044	9.384085	…
…	…	…	…	…	…	…	…

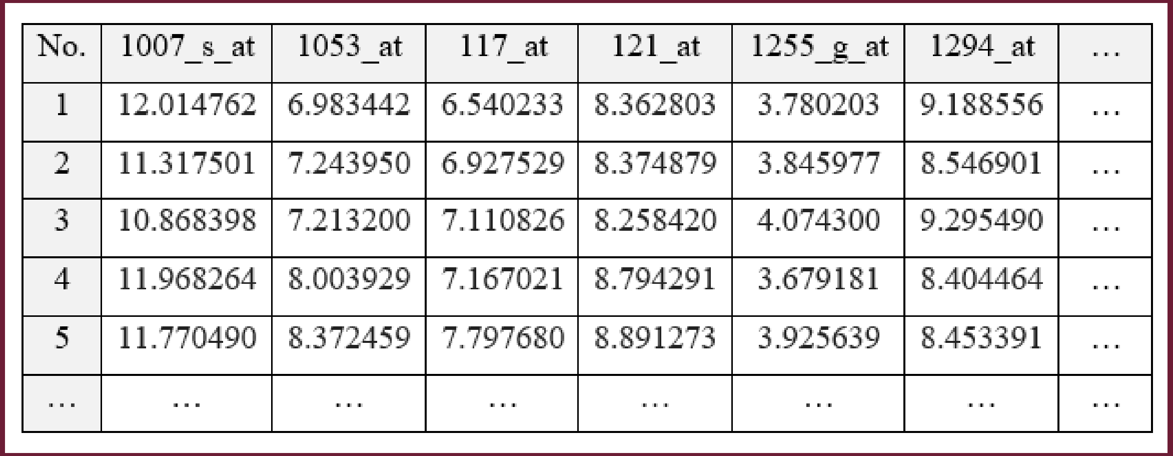
Colorectal Cancer Dataset After Normalization



No.	1007_s_at	1053_at	117_at	121_at	1255_g_at	1294_at	…
1	0.876381	0.173848	0.086729	0.506579	0.279423	0.747825	…
2	0.631176	0.176263	0.113178	0.842652	0.437781	0.601175	…
3	0.400561	0.167204	0.284344	0.353543	0.385999	0.605835	…
4	0.478749	0.299433	0.118443	0.746851	0.506567	0.870887	…
5	0.497174	0.024910	0.112750	0.848742	0.917759	0.766508	…
…	…	…	…	…	…	…	…

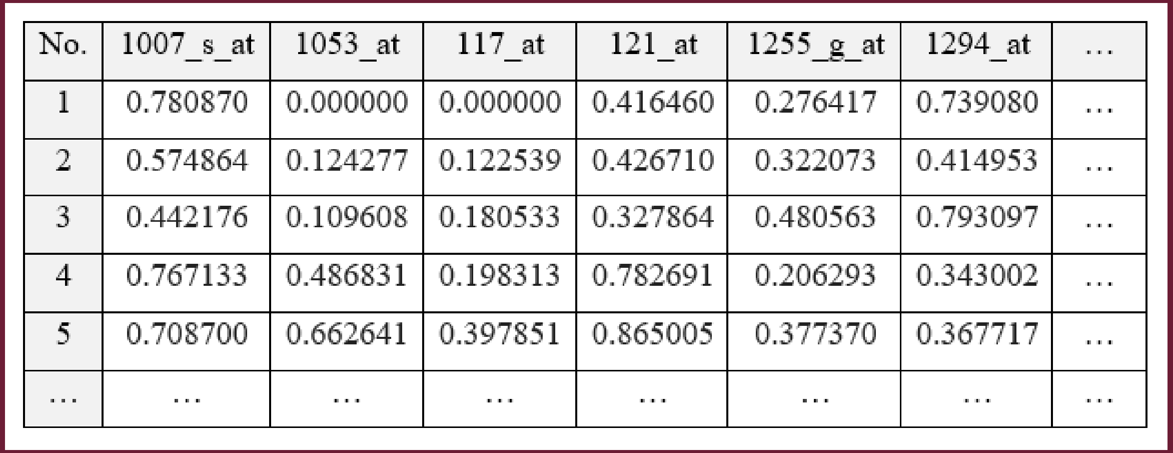
Data Normalization

Lung Cancer Dataset Before Normalization



No.	1007_s_at	1053_at	117_at	121_at	1255_g_at	1294_at	…
1	12.014762	6.983442	6.540233	8.362803	3.780203	9.188556	…
2	11.317501	7.243950	6.927529	8.374879	3.845977	8.546901	…
3	10.868398	7.213200	7.110826	8.258420	4.074300	9.295490	…
4	11.968264	8.003929	7.167021	8.794291	3.679181	8.404464	…
5	11.770490	8.372459	7.797680	8.891273	3.925639	8.453391	…
…	…	…	…	…	…	…	…

Lung Cancer Dataset After Normalization



No.	1007_s_at	1053_at	117_at	121_at	1255_g_at	1294_at	…
1	0.780870	0.000000	0.000000	0.416460	0.276417	0.739080	…
2	0.574864	0.124277	0.122539	0.426710	0.322073	0.414953	…
3	0.442176	0.109608	0.180533	0.327864	0.480563	0.793097	…
4	0.767133	0.486831	0.198313	0.782691	0.206293	0.343002	…
5	0.708700	0.662641	0.397851	0.865005	0.377370	0.367717	…
…	…	…	…	…	…	…	…

Data Splitting

Data Division for Colorectal Cancer

Phase	Data	Total
Training (80%)	1-84	84
Testing (20%)	85-105	21

Data Division for Lung Cancer

Phase	Data	Total
Training (80%)	1-91	91
Testing (20%)	92-114	23

Feature Selection

01 No Feature Selection

02 Chi-Square Feature Selection

03 SVM-RFE Feature Selection

04 Random Forest Feature Selection

05 PSO Feature Selection

Prefiltering Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`prefilter_n` (Selected top-ranked features)	100, 200, 500	**Colorectal:** 200 **Lung:** 100	To reduce computational complexity and search space before applying more intensive feature selection.	**Higher:** Keeps more genes, potentially more informative but slower. **Lower:** Reduces genes more aggressively, faster but risks losing important ones.
`score_func` (Scoring function)	Chi-Square, ANOVA F-test, T-test	Chi-Square	To identify the most effective statistical test for ranking genes during the prefiltering step, ensuring that the most informative features are retained for downstream analysis.	Comparing scoring functions helps choose the best way to find relevant genes, making data reduction more effective for later analysis.

No Feature Selection

- Used all features without selection

- No tuning needed

- Purpose:
  1. Provide performance benchmark
  2. Compare impact of feature selection

Chi-Square FS Tuning

Parameter	Range Tuned Value	Chosen Tuned Value	Why Tune the Parameter	Explanation
`k` (Number of features retained)	10, 30, 50, 70, 100	100	To find the optimal number of features for the best classification accuracy.	**Higher:** Includes more genes (can capture more patterns, but might be too complex). **Lower:** Focuses on fewer, most important genes (simpler model, less noise).

SVM-RFE FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`k` (Number of features retained)	10, 30, 50 70, 100	All	To find the optimal number of features for the best classification accuracy.	**Higher:** Includes more genes (can capture more patterns, but might be too complex). **Lower:** Focuses on fewer, most important genes (simpler model, less noise).
`step_size` (Elimination step size)	1, 5, 10, 20	All	To control how many features are removed at each step of the process.	**Larger:** Faster but might accidentally remove important features too quickly. **Smaller:** Slower but more precise.

SVM-RFE FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`C` (SVM Regularization Parameter)	0.01, 0.1, 1, 10, 100	All	To balance between fitting the training data perfectly and making a model that works well on new data.	**Higher:** Makes the model try to fit the training data very closely (risks overfitting). **Lower:** Makes the model simpler and better at generalizing to new data (less overfitting).
`kernel` (Kernel Type)	-	Linear Kernel	To allow the method to properly rank features and keep calculations straightforward.	A linear kernel is chosen because it allows for clear ranking of feature importance, which is essential for SVM-RFE.

Random Forest FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`k` (Number of features retained)	10, 30, 50 70, 100	**Colorectal:** 30 **Lung:** 10	To find the optimal number of features for the best classification accuracy.	**Higher:** Includes more genes (can capture more patterns, but might be too complex). **Lower:** Focuses on fewer, most important genes (simpler model, less noise).
`n_estimators` (Number of trees)	50, 100, 200	100, 200	To decide how many "decision trees" are built in the forest.	**More:** Give better, more stable predictions but take longer to compute. **Fewer:** Faster but might be less accurate.

Random Forest FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`max_depth` (Maximum depth)	None, 5, 10	All	To control how complex each individual decision tree can become.	**No limit (None):** Can lead to overfitting (memorizing training data). **Limiting depth:** Makes trees simpler and helps them generalize better to new data.
`min_samples_split` (Minimum samples to split)	2, 5	All	To set the minimum number of data points needed before a tree can split a node.	**Higher:** Lead to simpler trees that generalize better by avoiding tiny, noisy splits. **Lower:** Allow more detailed splits, potentially capturing fine patterns but risking overfitting.

Random Forest FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`min_samples_leaf` (Minimum samples at leaf)	1, 2	All	To control how complex each individual decision tree can become.	**Higher:** Provide more reliable leaves and help prevent overfitting. **Lower:** Allow leaves to be very specific to single data points, risking overfitting.
`max_features` (Number of features considered at each split)	sqrt, log2	All	To set the minimum number of data points needed before a tree can split a node.	Considering only a subset of features at each split makes the forest more diverse and robust, reducing overfitting and speeding up training.

PSO FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`k` (Number of features retained)	10, 30, 50 70, 100	10	To find the optimal number of features for the best classification accuracy.	**Higher:** Includes more genes (can capture more patterns, but might be too complex). **Lower:** Focuses on fewer, most important genes (simpler model, less noise).
`n_particles` (Number of particles)	10, 20, 30	10	To control how many "candidate solutions" (particles) are searching for the best set of genes. More particles mean a more thorough search.	**More:** Increase search diversity (better solutions, but slower). **Fewer:** Faster (but less thorough).

PSO FS Tuning

Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`c1` (Cognitive coefficient)	1.5, 2.0, 2.5	1.5	To control how much a particle sticks to its own best-found path.	**Higher:** Means particles follow their own past success more. **Lower:** Allows more influence from the group or new exploration.
`c2` (Social coefficient)	1.5, 2.0, 2.5	**Colorectal:** 1.5, 2.0 **Lung:** 1.5	To control how much a particle follows the best path found by the entire group.	**Higher:** Means particles are more influenced by the group's best. **Lower:** Means less group influence, more individual search.

PSO FS Tuning

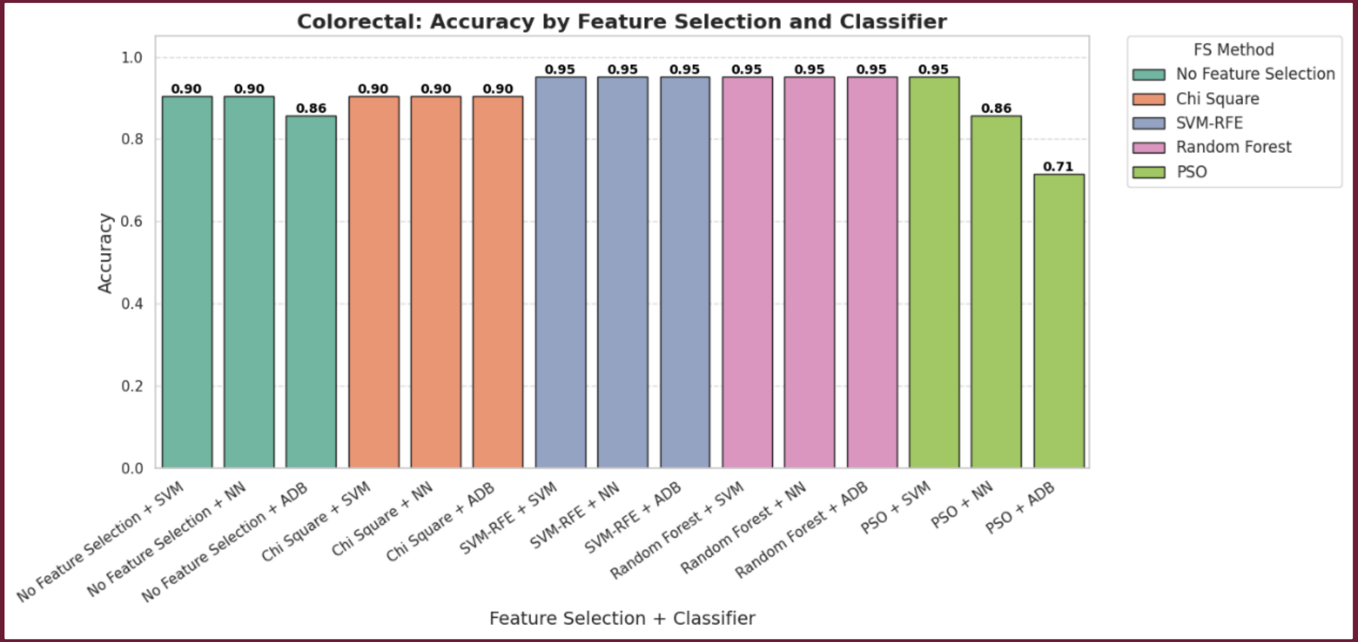
Parameter	Range Tuned Value/Test	Chosen Tuned Value	Why Tune the Parameter	Explanation
`w` (Inertia weight)	0.7, 0.9	All	To balance exploring new areas versus refining current promising ones.	**Higher:** Encourages exploration (finds new areas). **Lower:** Focuses on refining current solutions.
`max_iter` (Number of iterations)	20, 30, 50	All	To determine how long the search for the best gene set continues.	**Higher:** Allow for better refinement (higher quality, but slower). **Fewer:** Faster (but may stop too early).

CHAPTER 5
RESULTS

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

Results:
Colorectal Dataset

Accuracy (Colorectal)

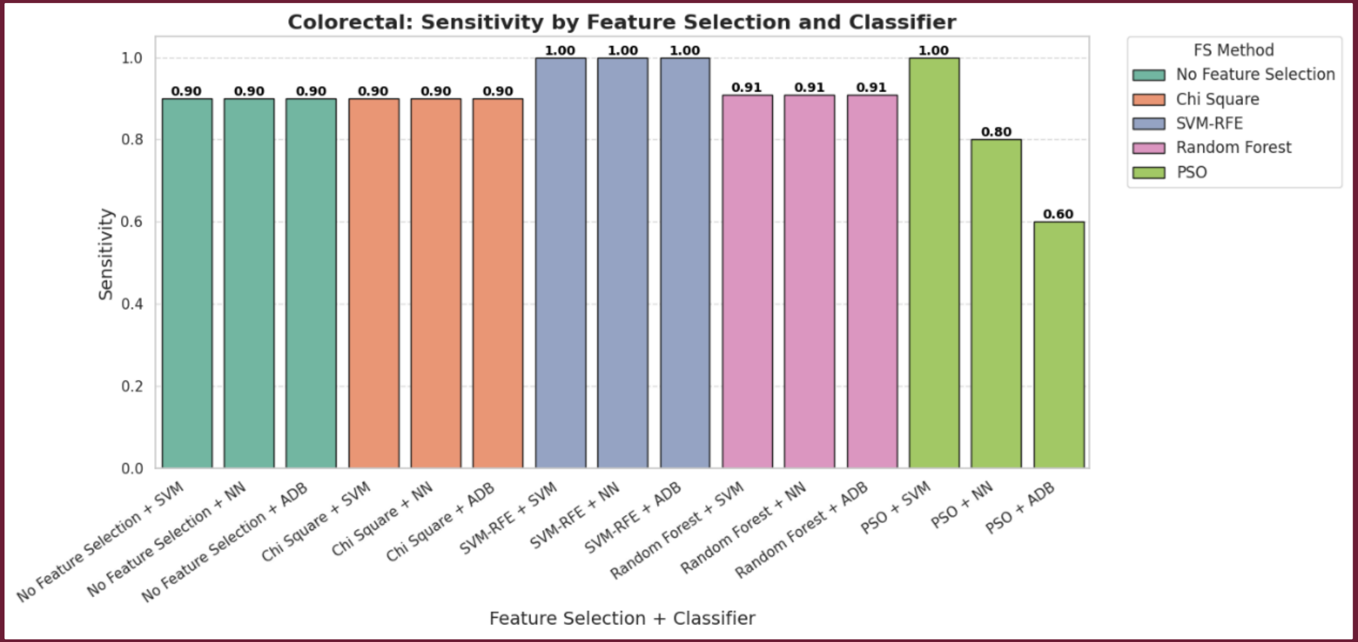


Highest Accuracy (0.95): Achieved by SVM-RFE and Random Forest with various classifiers, and PSO + SVM.

Lowest Accuracy (0.71): Observed with PSO combined with AdaBoost.

***Key:** Method choice significantly impacts performance accuracy.

Sensitivity (Colorectal)



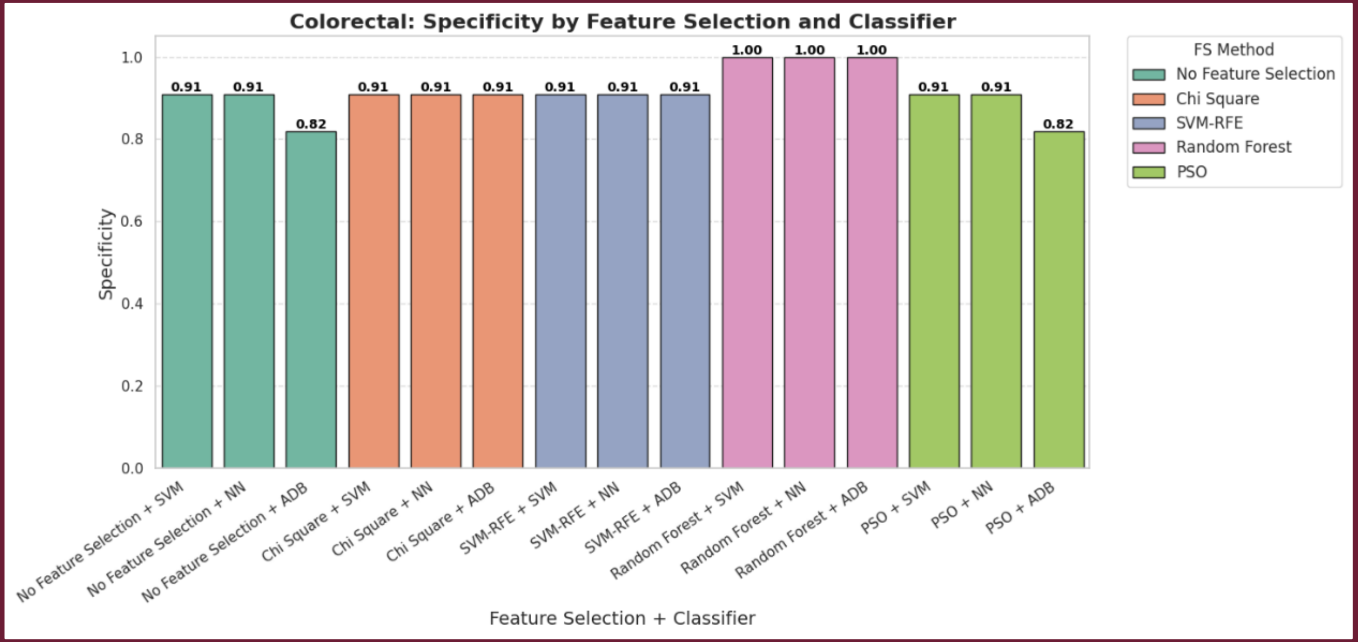
Perfect Sensitivity (1.00): Achieved by SVM-RFE (all classifiers) and PSO + SVM.

Significant Drop (0.60): PSO combined with AdaBoost showed a notable decrease in detecting true positive cases.

Other Methods: Remained stable around 0.90–0.91.

***Key:** SVM-RFE is highly reliable for identifying actual positive cases, crucial for medical diagnosis. PSO + AdaBoost performed poorly in this aspect.

Specificity (Colorectal)



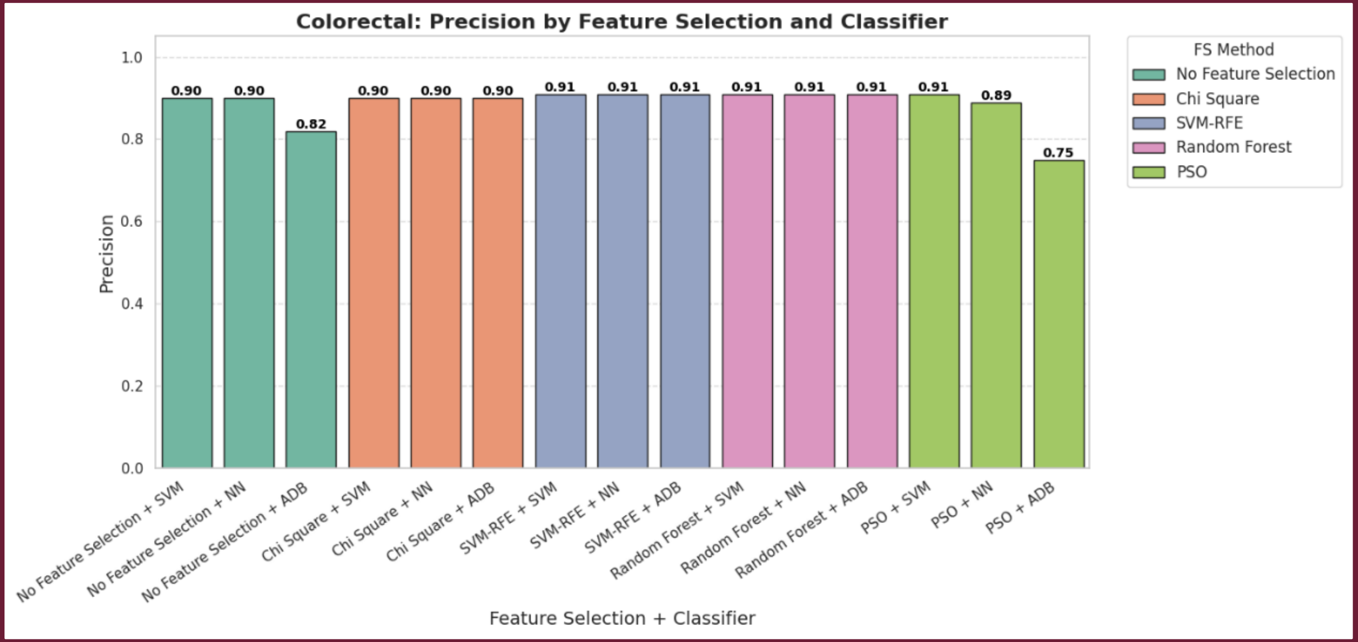
Perfect Specificity (1.00): Achieved by all Random Forest combinations (SVM, NN, AdaBoost).

Lowest Specificity (0.82): Seen with PSO + AdaBoost and No Feature Selection + AdaBoost (indicating more false positives).

Most Others: Maintained high specificity (~0.91).

***Key:** Random Forest excels at avoiding false alarms, while PSO + AdaBoost is less effective at distinguishing negative cases.

Precision (Colorectal)



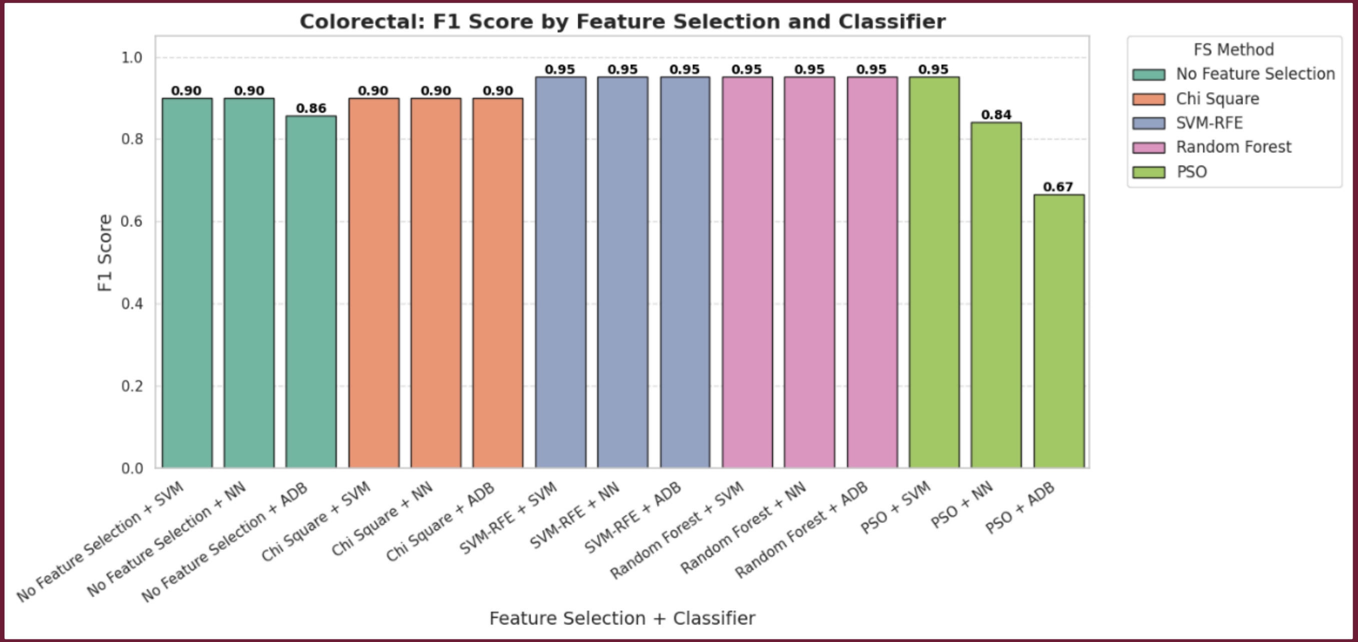
High Precision (0.91): Achieved by SVM-RFE and Random Forest with various classifiers, and PSO + SVM.

Lowest Precision (0.75): Seen with PSO + AdaBoost.

Most Others: Remained stable around 0.90–0.91.

***Key:** SVM-RFE, Random Forest, and PSO + SVM show strong reliability in correctly identifying positive cases; PSO + AdaBoost performs considerably worse in this aspect.

F1-Score (Colorectal)



High F1-Score (0.95): Achieved by SVM-RFE and Random Forest with various classifiers, and PSO + SVM.

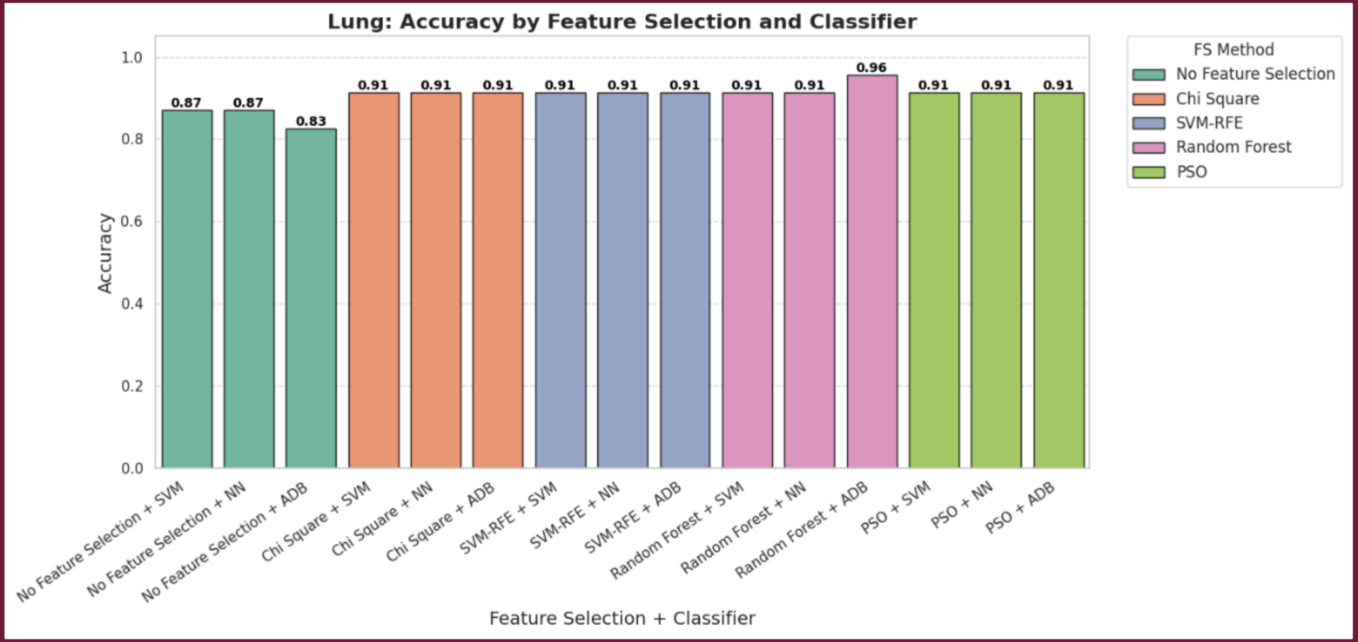
Other F1-Scores: No Feature Selection + AdaBoost (0.86), PSO + Neural Network (0.84). Most others maintained ~0.90.

Lowest F1-Score (0.67): Observed with PSO + AdaBoost.

***Key:** SVM-RFE, Random Forest, and PSO with SVM show strong balanced performance; PSO + AdaBoost consistently struggles.

Results:
Lung Dataset

Accuracy (Lung)



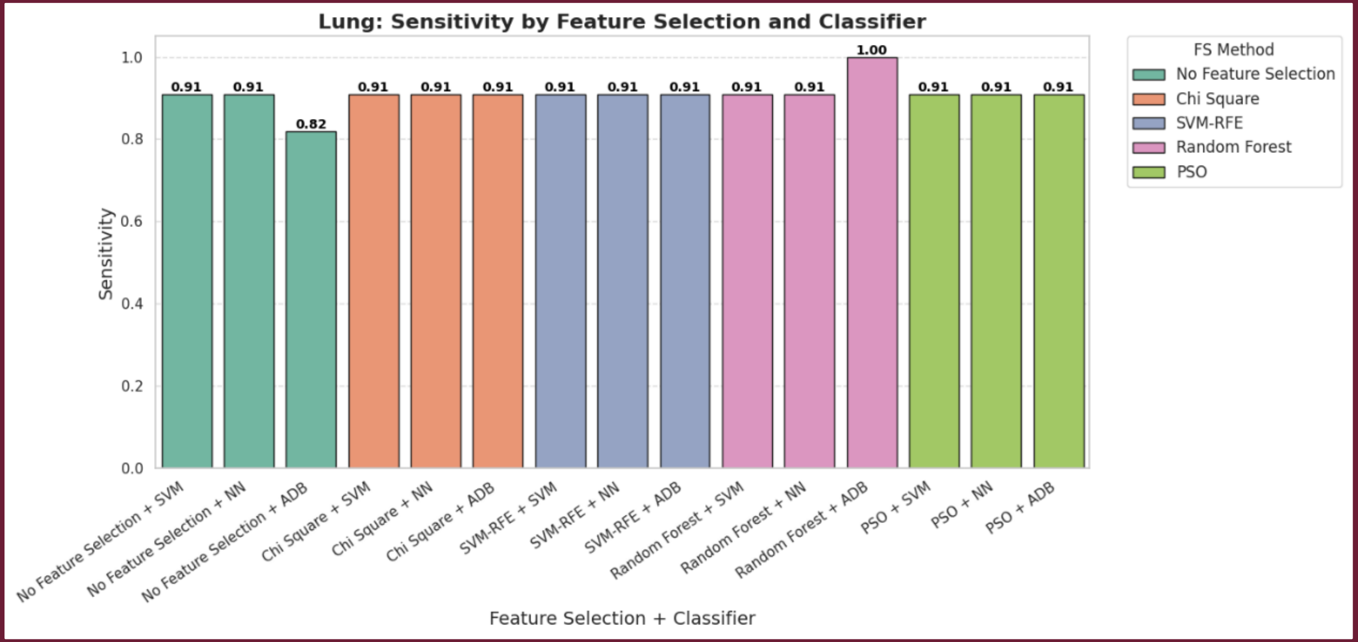
Highest Accuracy (0.96): Achieved by Random Forest + AdaBoost.

Other Accuracy: No Feature Selection + SVM/NN (0.87); most others maintained 0.91.

Lowest Accuracy (0.83): Observed with No Feature Selection + AdaBoost.

***Key:** Random Forest + AdaBoost delivered the highest overall correct predictions, while combinations without feature selection performed less optimally.

Sensitivity (Lung)



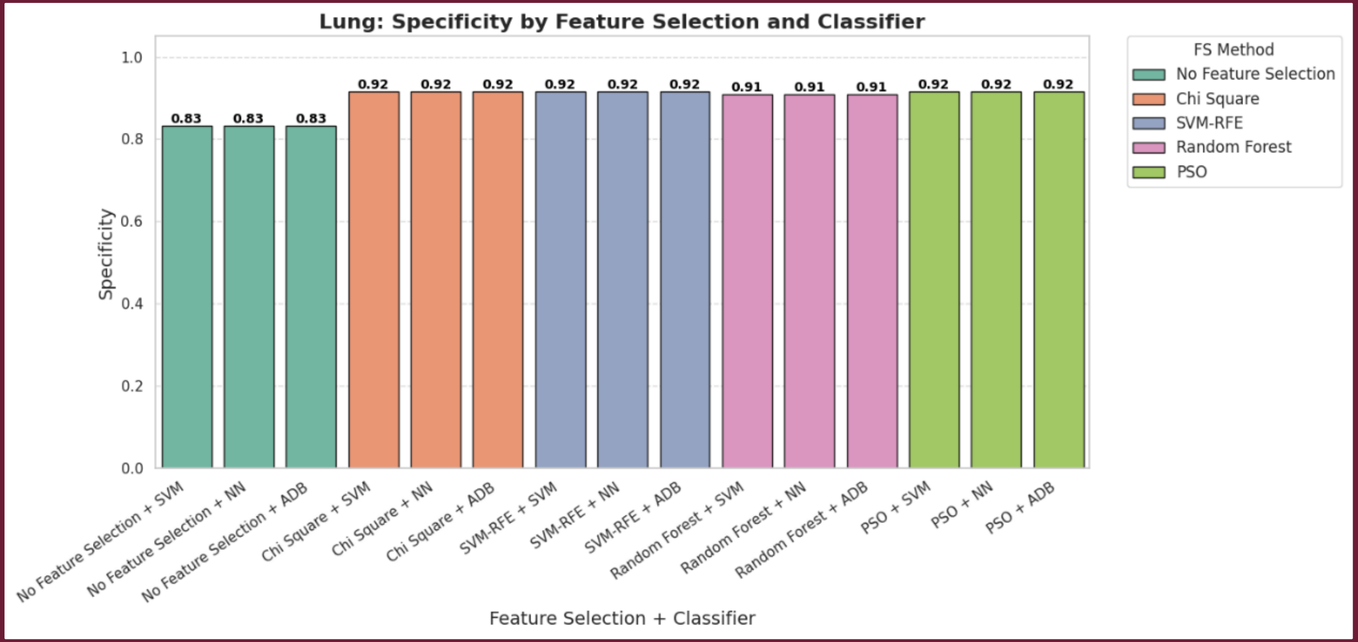
Perfect Sensitivity (1.00): Achieved by Random Forest + AdaBoost.

Other Sensitivity: Most others maintained ~0.91.

Lowest Sensitivity (0.82): Observed with No Feature Selection + AdaBoost.

***Key:** Random Forest + AdaBoost achieved perfect sensitivity (1.00), effectively detecting all positive cases, while lacking feature selection with AdaBoost yielded the lowest sensitivity (0.82), indicating more missed diagnoses.

Specificity (Lung)



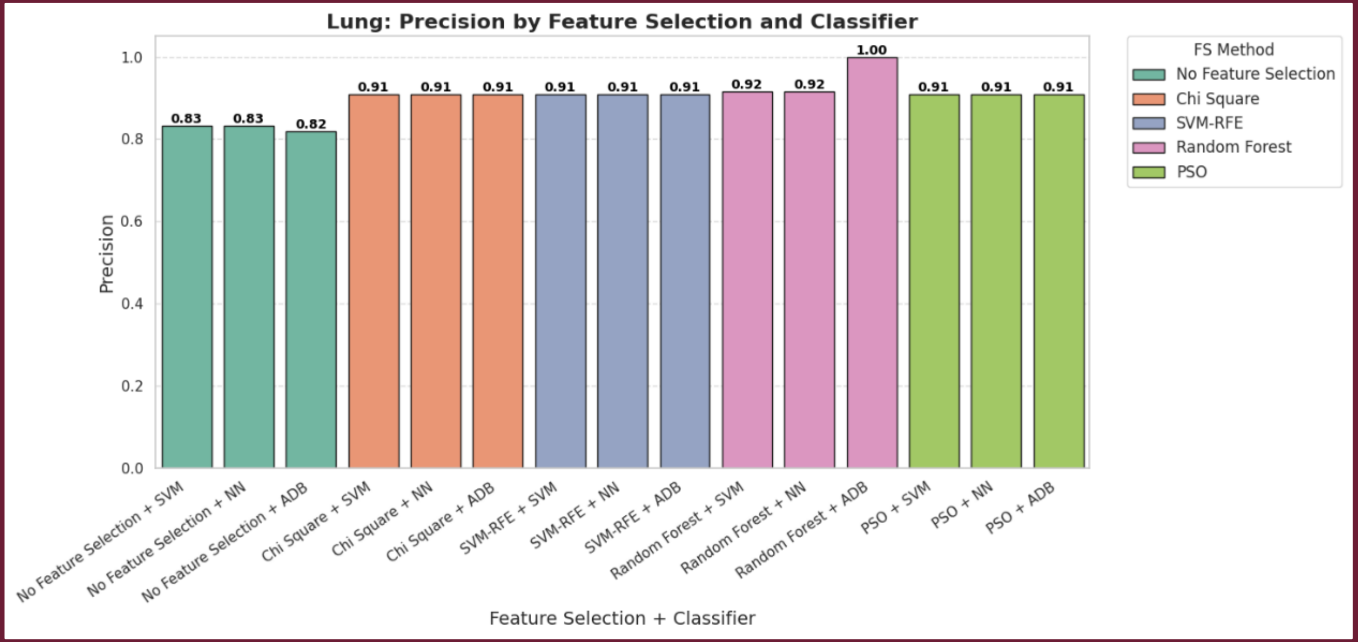
Highest Specificity (0.92): Achieved by Chi-Square, SVM-RFE, and PSO with all classifiers.

Middle Specificity (0.91): Observed with Random Forest (all classifiers).

Lowest Specificity (0.83): Observed with No Feature Selection (all classifiers).

***Key:** Feature selection, especially with Chi-Square, SVM-RFE, or PSO, significantly improves the correct identification of healthy cases, minimizing false alarms.

Precision (Lung)



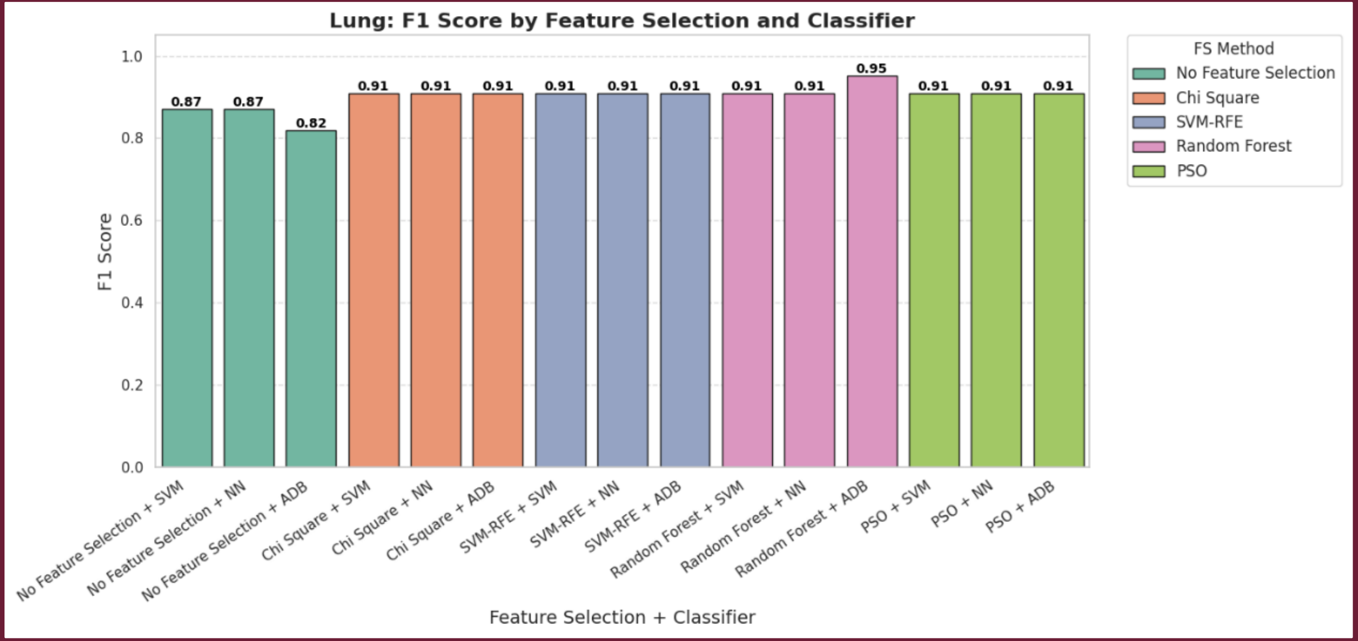
Perfect Precision (1.00): Achieved by Random Forest + AdaBoost.

Other Precision: Random Forest + SVM/NN (0.92), mostly ~0.91, No Feature Selection + SVM/NN (0.83)

Lowest Precision (0.82): Observed with No Feature Selection + AdaBoost.

***Key:** Random Forest + AdaBoost provides the most reliable positive predictions, while the absence of feature selection, particularly with AdaBoost, results in less trustworthy positive diagnoses.

F1-Score (Lung)



Highest F1-Score (0.95): Achieved by Random Forest + AdaBoost.

Other F1-Score: Mostly ~0.91; No Feature Selection + SVM/NN (0.87)

Lowest F1-Score (0.82): Observed with No Feature Selection + AdaBoost.

***Key:** Random Forest + AdaBoost consistently delivers the strongest balanced performance, while no feature selection (especially with AdaBoost) leads to the lowest F1-score.

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 6
CONCLUSION

OPTIMIZING CANCER GENE EXPRESSION DATA CLASSIFICATION PERFORMANCE THROUGH PARTICLE SWARM OPTIMIZATION (PSO)

Key Conclusion

- PSO is effective in certain settings (lung cancer + SVM).
- However, classifier compatibility and dataset type matter.
- No one-size-fits-all solution—careful method selection is important.

Research Constraints

01

PSO performance inconsistency across classifiers.

02

High computational cost for tuning (especially SVM-RFE, PSO).

03

Limited datasets (colorectal & lung only).

Future Work

01

Apply to more diverse cancer datasets such multi-class.

02

Use hybrid PSO versions (PSO-GA, fuzzy PSO, chaotic PSO).

03

Explore deep learning and ensemble classifiers for further performance gains.

THANK YOU

univteknologimalaysia utm.my utmofficial