

# Algorithm Description: Auto Feature Selection and Model Training Pipeline

---

**Algorithm 1** Auto Feature Selection and Model Training Pipeline

---

```
1: Class EnsembleForRFE(BaseEstimator)
2:   Input: svm_C, tree_max_depth, tree_min_samples_split, gbm_learning_rate, gbm_n_estimators
3:   Operations:
4:     Initialize SVM, Decision Tree, and Gradient Boosting with specified parameters
5:     Define methods: fit(X, y), predict(X), set_params(**params)
6:   End Class
7:
8: Function setup_feature_selection()
9:   Operations:
10:    Create an instance of EnsembleForRFE
11:    Initialize RFECV with EnsembleForRFE and StratifiedKFold
12:    Initialize SelectKBest with mutual_info_classif
13:    Combine RFECV and SelectKBest using FeatureUnion
14:   Return: feature_selection
15:
16: Function train_model(X, Y, feature_selection, parameters, n_iter, n_cv, n_jobs)
17:   Operations:
18:    Configure a pipeline with StandardScaler, feature_selection, and StackingClassifier
19:    Define RandomizedSearchCV with the pipeline and specified parameters
20:   Return: clf (trained model)
21:
22: Function auto_feature_selection(data_file, label_file, label_col, threshold, show_plot,
    show_progress, n_iter, n_cv, n_jobs, save_path, sleep_interval, use_tkagg)
23:   Operations:
24:    Load and preprocess data from data_file and label_file
25:    Call setup_feature_selection() to configure feature selection
26:    Define parameter distribution for RandomizedSearchCV
27:    Train the model using train_model()
28:    if show_progress then
29:      Initialize and manage a progress bar
30:      Run RandomizedSearchCV in a separate thread and monitor progress
31:    else
32:      Directly execute RandomizedSearchCV
33:      Extract and save results, display plots if required
34:   End Function
```

---

This document illustrates a complete pipeline from data loading, preprocessing, feature selection setup, model training with hyperparameter tuning, to result extraction and evaluation. The progression from data setup to final results encapsulates the complexity and interdependencies of the involved components.