🔬Melting Point Prediction using Two-Level Ensemble Learning

This project predicts the melting points of organic compounds using a robust two-layer ensemble machine learning model. It combines multiple base learners with a Gradient Boosting meta-model, enabling accurate and generalizable predictions on both validation and unseen molecular data.

📁Dataset & Features

- **Inputs**: Molecular SMILES strings processed using RDKit.
- **Output**: Experimentally measured melting points (°C).
- **Featurization**: Molecular descriptors via RDKit; optionally expandable with Matminer or Mendeleev.
- **Splits**: Train, validation, and an independent test set.

Model Architecture

🕰️First Layer: Base Learners

An ensemble of 10 models in a custom sequence:

```
['gb', 'rf', 'mlp', 'rf', 'gb', 'mlp', 'gb', 'rf', 'mlp', 'gb']
```

- `gb`: Gradient Boosting Regressor (Bayesian-optimized)
- `rf`: Random Forest Regressor
- `mlp`: Multi-layer Perceptron with variable depth

Each model is trained on bootstrapped data with random feature subsets to enhance diversity.

🕰️Second Layer: Stacking Regressor

A **Gradient Boosting Regressor** is used as a meta-learner trained on predictions from the base layer.

⚙️Optimization & Reproducibility

- **Hyperparameter Tuning**: Conducted using `BayesSearchCV` from `scikit-optimize`.
- **MLP Sensitivity Analysis**: Best performance with 5 hidden layers of 200 neurons each.
- **Reproducibility**: Controlled with consistent `random_state`, fixed seeds, and thread restrictions.

📈Performance

Metric	Validation Set	Unseen Test Set
R^2 Score	**0.831**	**0.830**
MAE (°C)	**29.32**	**29.55**

📉**Predicted vs True** plots confirm tight correlation.

📦Exported Model

The full model stack (base models + stacker) is saved via `joblib`:

```
joblib.dump(model_stack_unseen, 'melting_point_model_stack.pkl')
```

This can be loaded for future predictions on new compounds.

🧪How to Use

1. Install requirements: `rdkit`, `scikit-learn`, `joblib`, `matplotlib`, `scikit-optimize`
2. Run the main notebook.
3. Evaluate and visualize results with `y_unseen` + `X_test_final`.

Author & Acknowledgments

This work is inspired by ensemble methods in cheminformatics literature, particularly the stacking techniques of Kiselyova et al. and Senko et al.