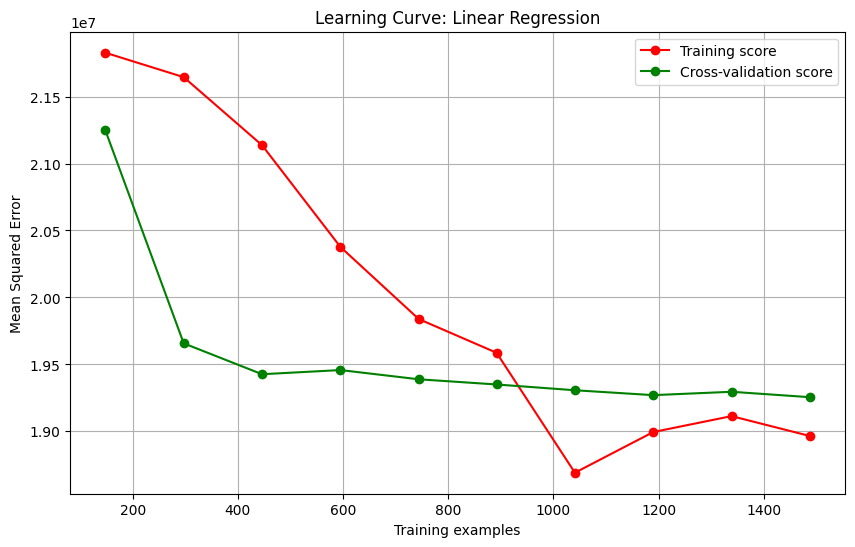
# 1. Data preprocessing

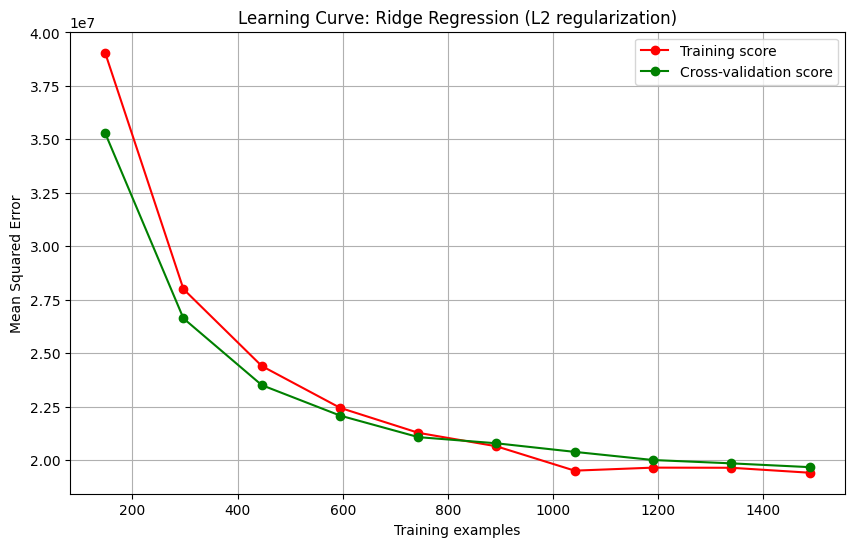
# 2. Build ML models

# 3. Evaluate overfitting / underfitting for the eight ML methods

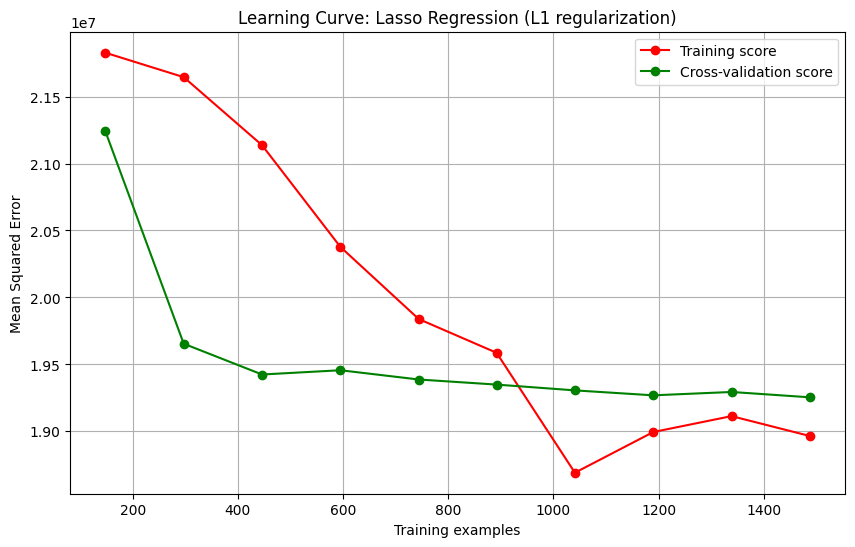
Generating learning curves...

****

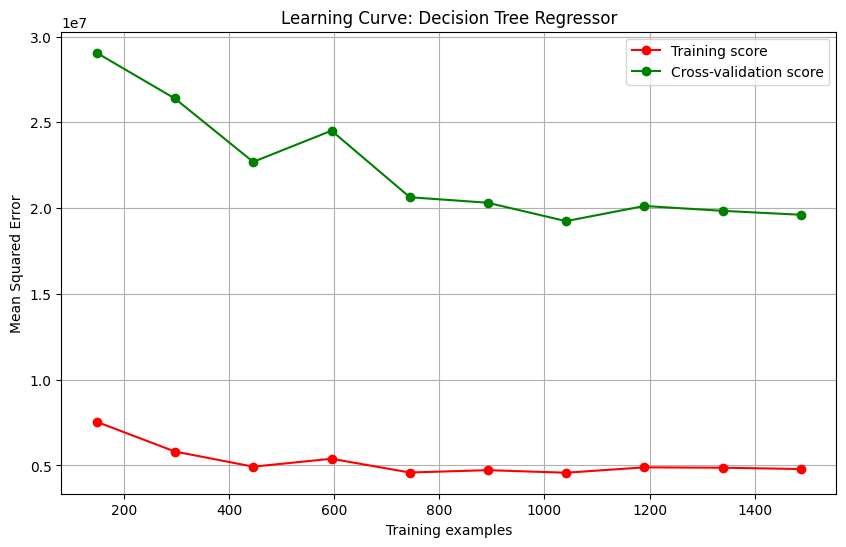
✓ Successfully saved: plots/Linear\_Regression.png

****

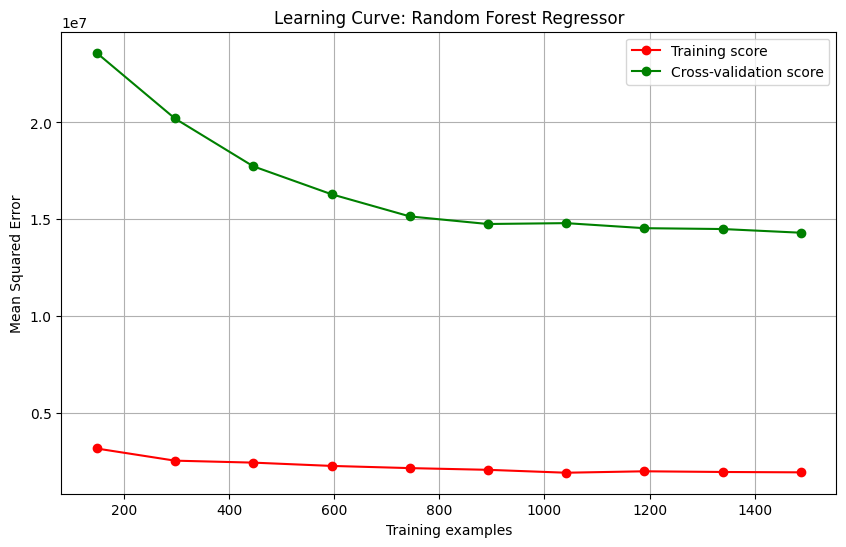
✓ Successfully saved: plots/Ridge\_Regression\_L2.png

****

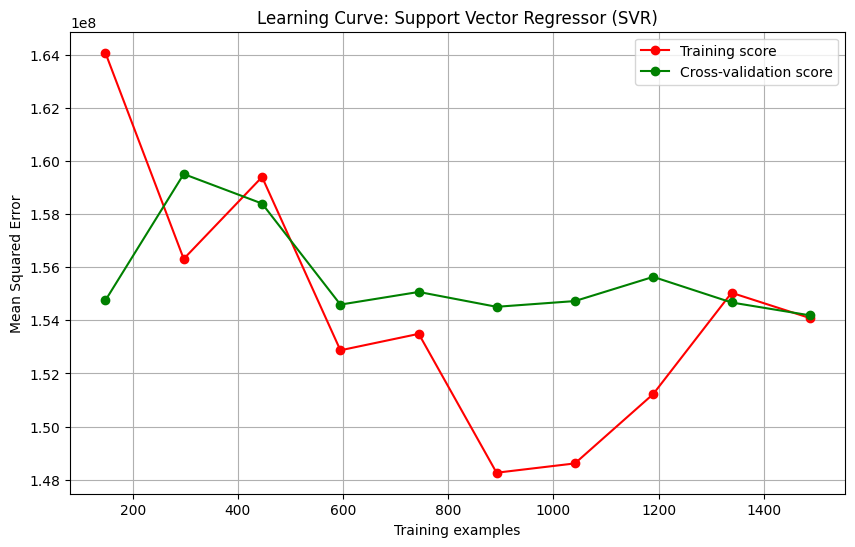
✓ Successfully saved: plots/Lasso\_Regression\_L1.png

****

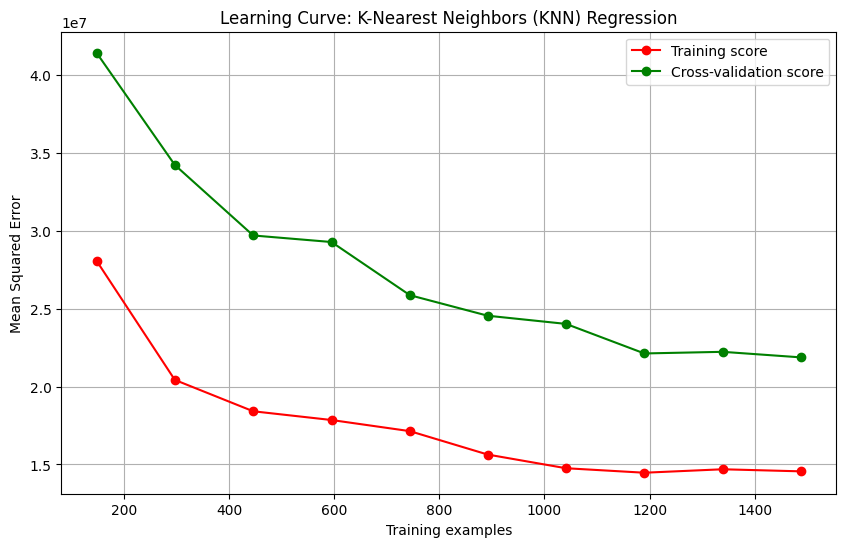
✓ Successfully saved: plots/Decision\_Tree\_Regressor.png

****

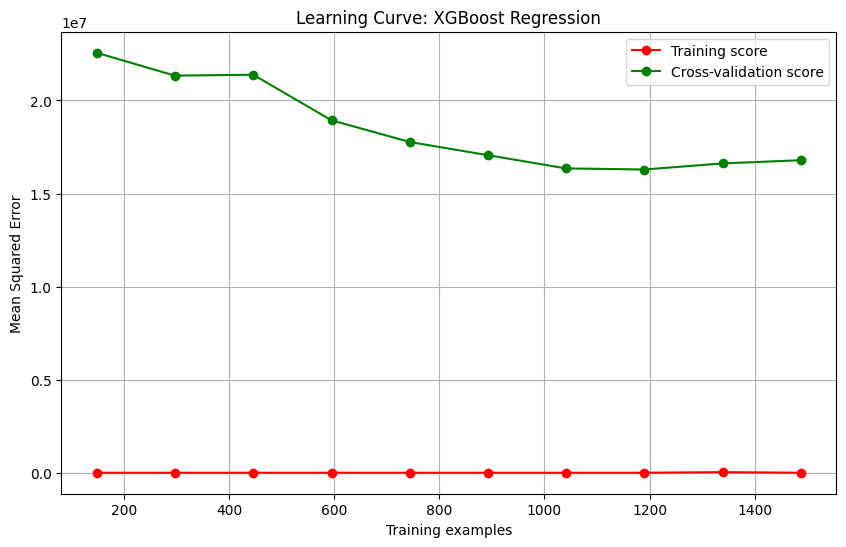
✓ Successfully saved: plots/Random\_Forest\_Regressor.png

****

✓ Successfully saved: plots/Support\_Vector\_Regressor\_SVR.png

****

✓ Successfully saved: plots/KNN\_Regression.png

****

✓ Successfully saved: plots/XGBoost\_Regression.png

All plots generated!

# 4. Hyperparameters tunning

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Tuning Lasso Regression...

Fitting 5 folds for each of 10 candidates, totalling 50 fits

Lasso - Best parameters: {'alpha': np.float64(46.41588833612773)}

Lasso - Best CV score: -19231998.9332

Lasso - Test set R² score: 0.8357

Lasso - Test set MSE: 23429492.2760

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Tuning Ridge Regression...

Fitting 5 folds for each of 10 candidates, totalling 50 fits

Ridge - Best parameters: {'alpha': np.float64(2.154434690031882)}

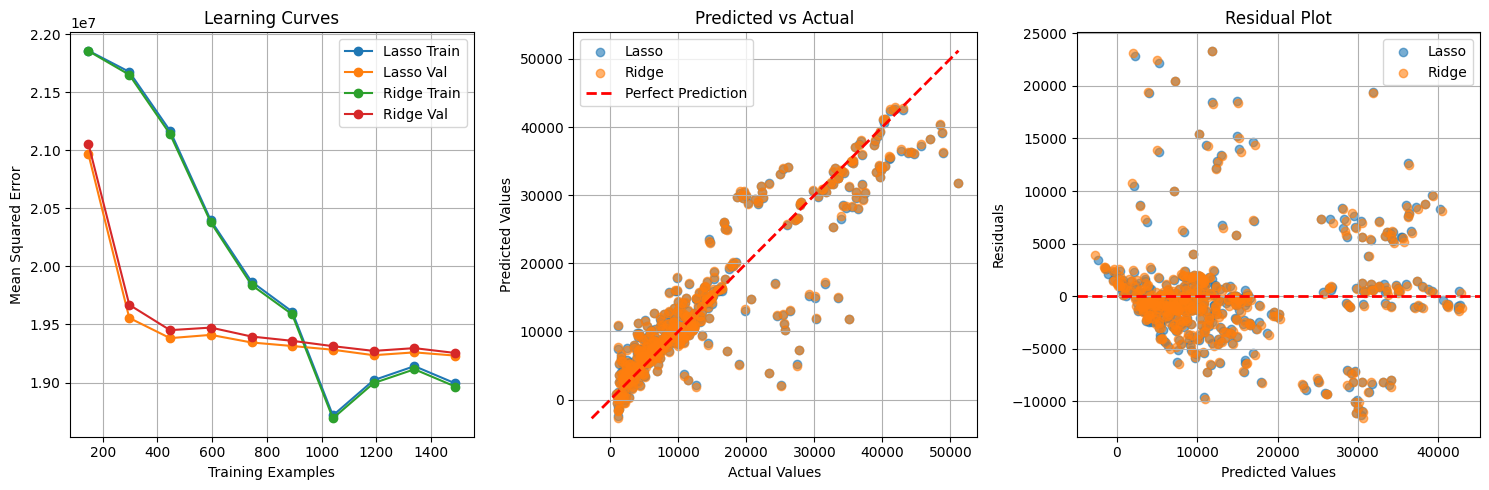
Ridge - Best CV score: -19254493.9437

Ridge - Test set R² score: 0.8359

Ridge - Test set MSE: 23403632.4924

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# 5. Performance comparison between ML models

****

Model Performance Metrics:

==================================================

Lasso | MSE: 23429492.2760 | R²: 0.8357

Ridge | MSE: 23403632.4924 | R²: 0.8359

# 6. Save the best model

Model saved successfully!