KI Service Zentrum
by Hasso-Plattner-Institut

HPI

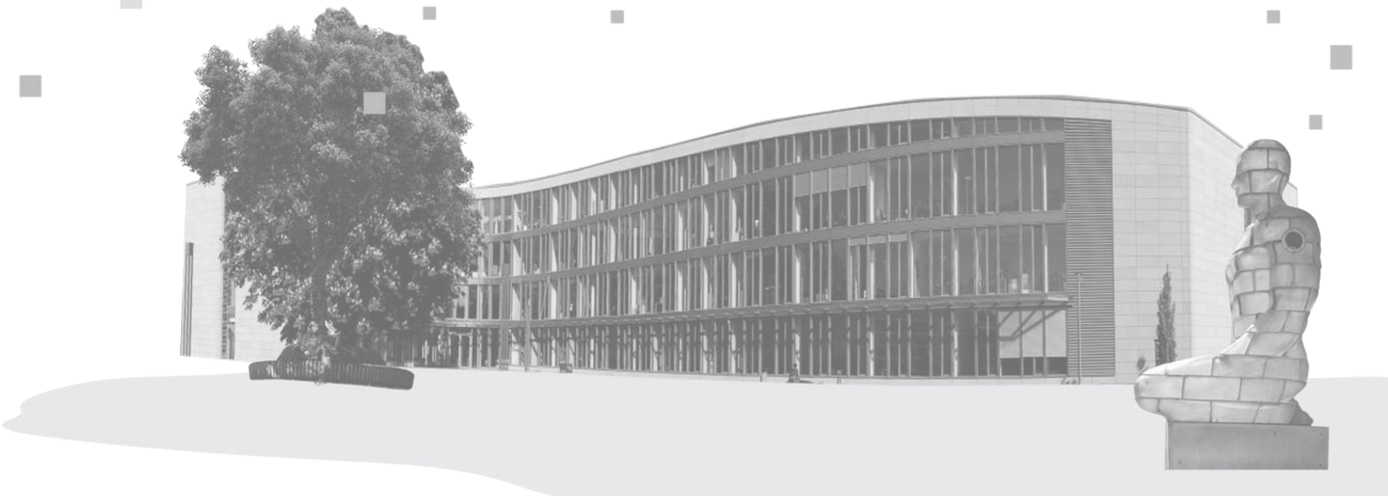
Gefördert durch:

Bundesministerium
für Forschung, Technologie
und Raumfahrt

Time Series Forecasting

2.3 Machine learning models

Mario Tormo Romero



Design IT.
Create Knowledge.

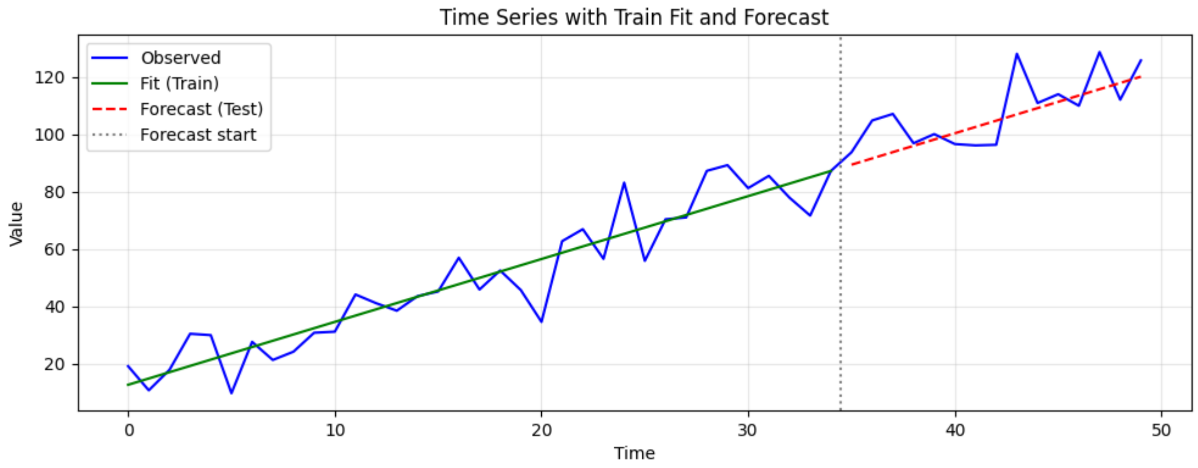
www.hpi.de

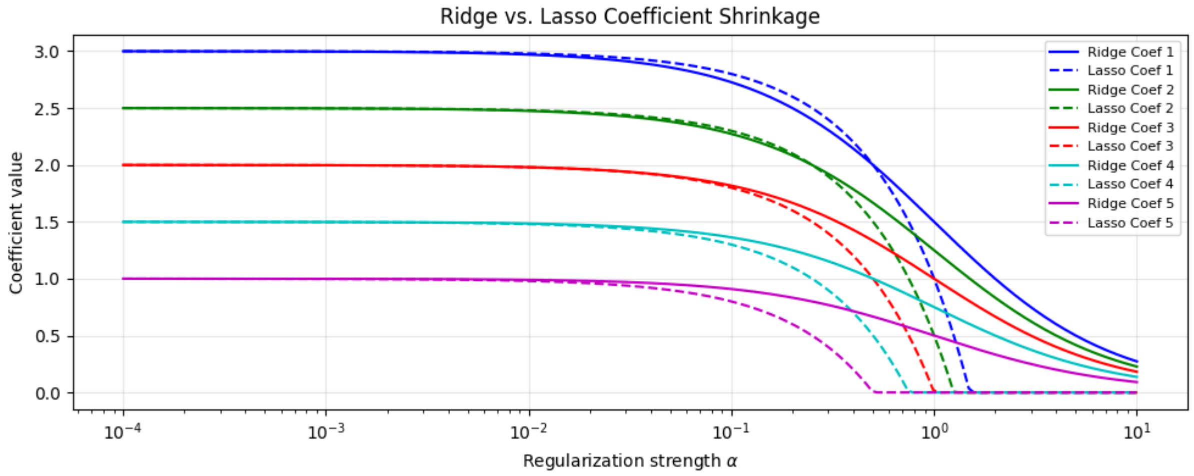
What we‵ll cover in this video

- How linear regression works for forecasting time series
- Why regularization (Ridge & Lasso) helps prevent overfitting
- Using decision trees to capture non-linear patterns
- How random forests improve stability with ensembles
- The power of gradient boosting decision trees (XGBoost, LightGBM, CatBoost)
- When to choose each model for your forecasting task

Linear regression and Regularization

- Linear regression models the relationship between features (lags, rolling stats, etc.) and the target.
- Regularization helps prevent overfitting by penalizing large coefficients.
  - **Ridge (L2)**: Shrinks coefficients toward zero but never exactly zero.
  - **Lasso (L1)**: Can shrink some coefficients exactly to zero (feature selection).
- Works best when the relationship is roughly linear and features are well-prepared.

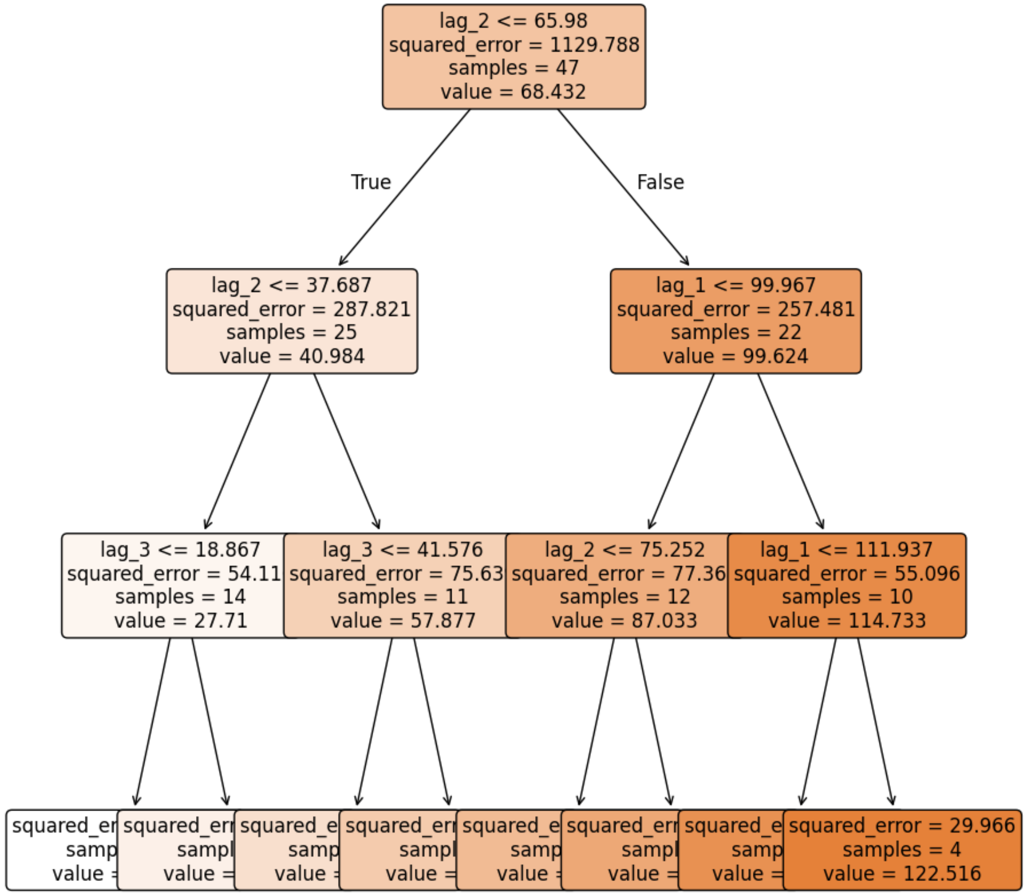




Decision trees & Random forests

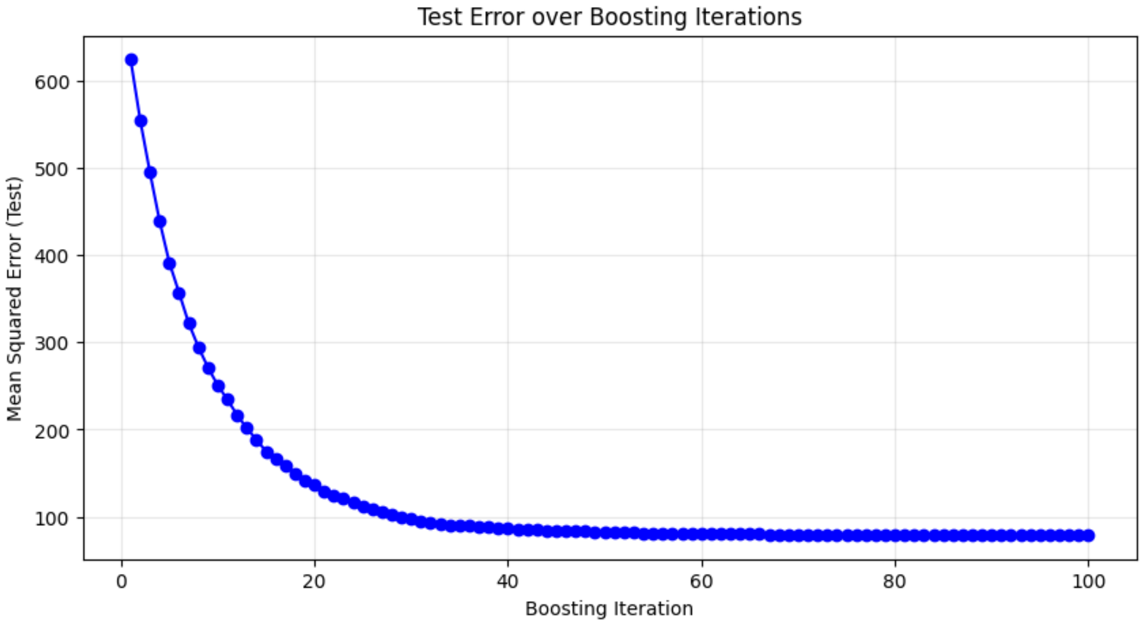
- **Decision Trees**:
  - Split data by feature thresholds to model non-linear relationships
  - Easy to visualize and interpret
  - Can overfit on training data if grown too deep
- **Random Forests**:
  - Ensemble of many decision trees trained on random subsets of data and features
  - Reduces overfitting and improves stability
  - Handles complex patterns and interactions without much feature engineering

Decision Tree trained on lagged Time Series Features



Gradient Boosted Trees

- Builds models sequentially, each new tree correcting errors from the previous ones
- Unlike random forests, which train trees independently in parallel, boosting builds trees one after another, focusing on previous errors
- Powerful at capturing complex, non-linear patterns
- Popular implementations: **XGBoost**, **LightGBM**, **CatBoost**
- Often achieves better accuracy than random forests but requires careful hyperparameter tuning to avoid overfitting



Model Use Cases & Selection Tips

- **Linear Regression**:
  - When you want a fast, interpretable baseline
  - Best if relationships are roughly linear and features are clean

- **Decision Trees & Random Forests**:
  - Great for capturing non-linear patterns and interactions
  - Random forests add stability and reduce overfitting
  - Useful when you have mixed types of features or missing data

- **Gradient Boosted Trees**:
  - When you need top predictive performance
  - Works well with complex patterns and large feature sets
  - Requires more tuning and validation time

What we‘ve learnt

- Linear regression gives a simple, interpretable baseline for forecasting
- Decision trees capture non-linear patterns; random forests improve stability
- Gradient boosted trees build models sequentially for powerful, accurate predictions
- Each model has strengths and trade-offs; tuning and validation are key
- Choosing the right model depends on your data and forecasting goals