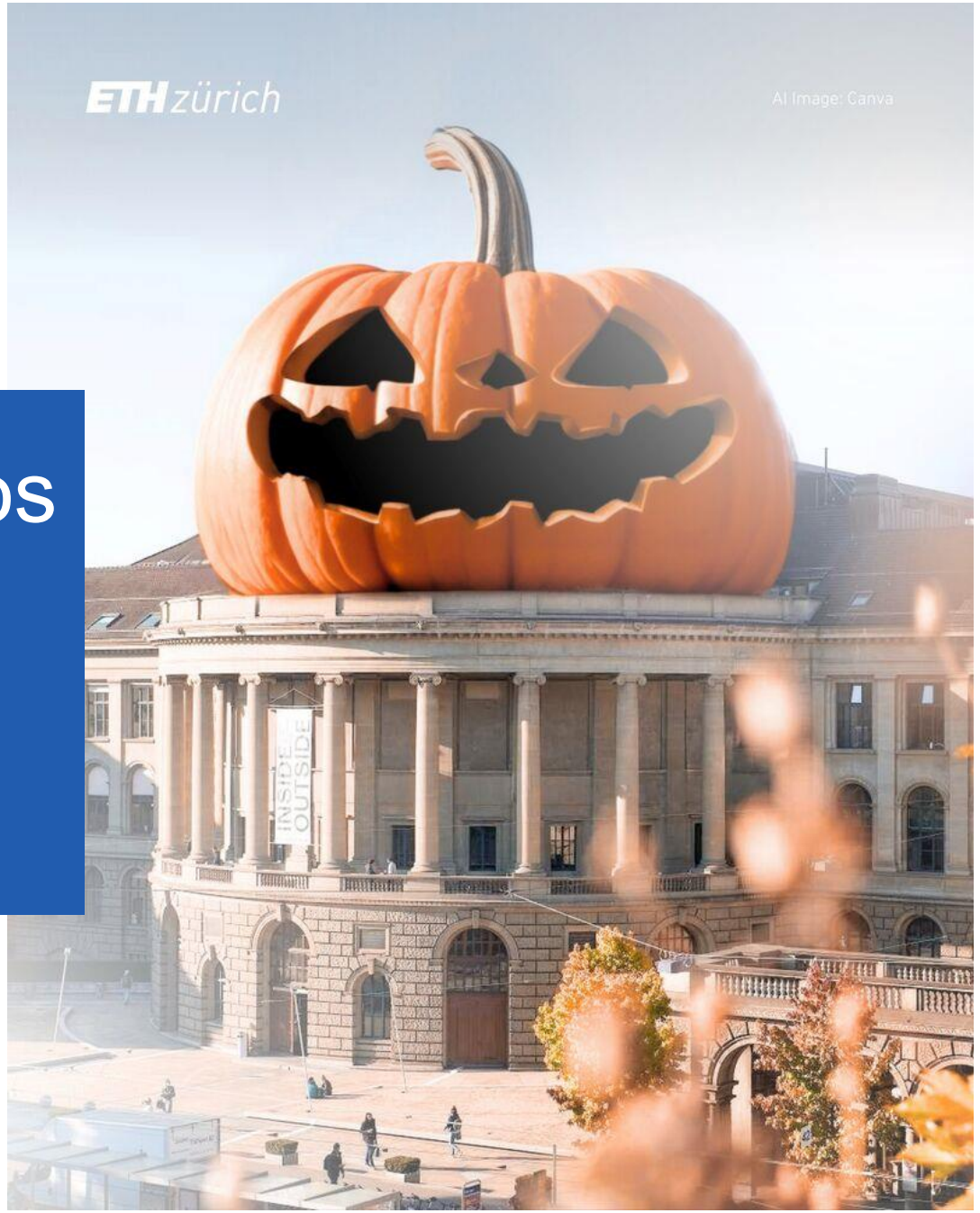
Estimating Solvency Ratios Using Machine Learning

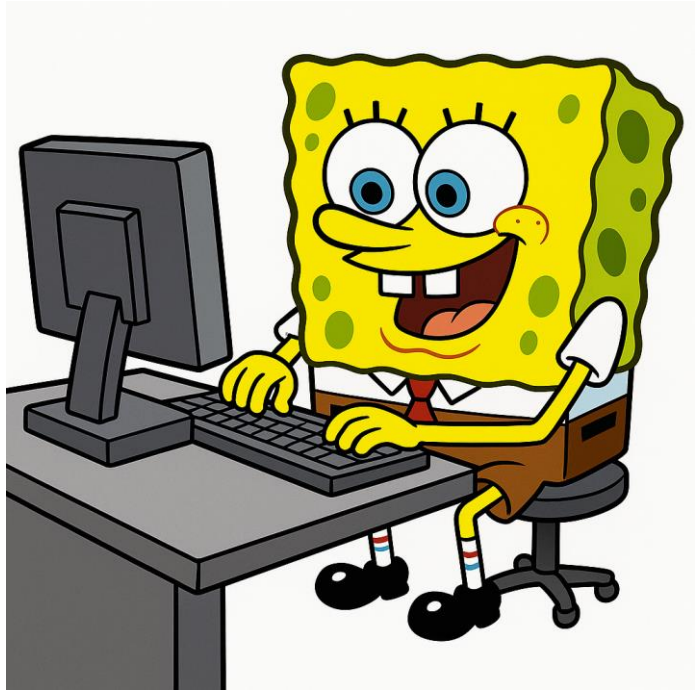
Oliver Stoll
CAS Innovation Project
07.11.2025



We want to forecast solvency ratios for life insurance over a short time horizon

- Solvency II / Swiss Solvency Test: reporting regime in the EU and Switzerland for insurance companies
- **Solvency ratio** $= \frac{\text{Available capital}}{\text{Required capital}} \gg 100\%$

- Risk management: **continuous** monitoring of solvency situation
- **Problem**: Full actuarial runs are slow & costly
- **Idea**: Use ML tools to learn **dependencies** between **economic market conditions** and **solvency ratio** *
- **Solution**: Feed new market parameters into ML model and instantly get a new solvency estimate



* Underlying assumption: short-term movements of ratio mainly driven by external market rates

Generating the synthetic input data required a lot of effort

Input data
- 10’230 rows x 20 features $\in [0,1]^{20}$:
  - 8 market + 12 flags/rules

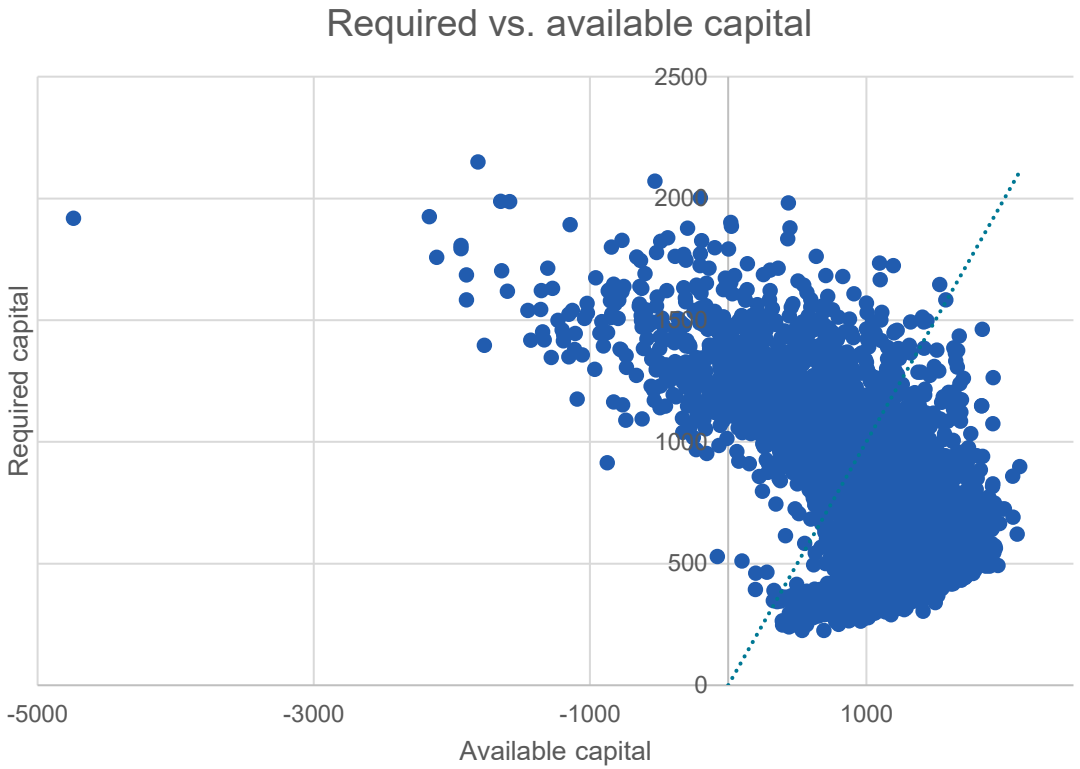
ESG *
- Calibrate economic scenarios
  - Baseline + shocks

Runs
- Stochastic asset-liability projections

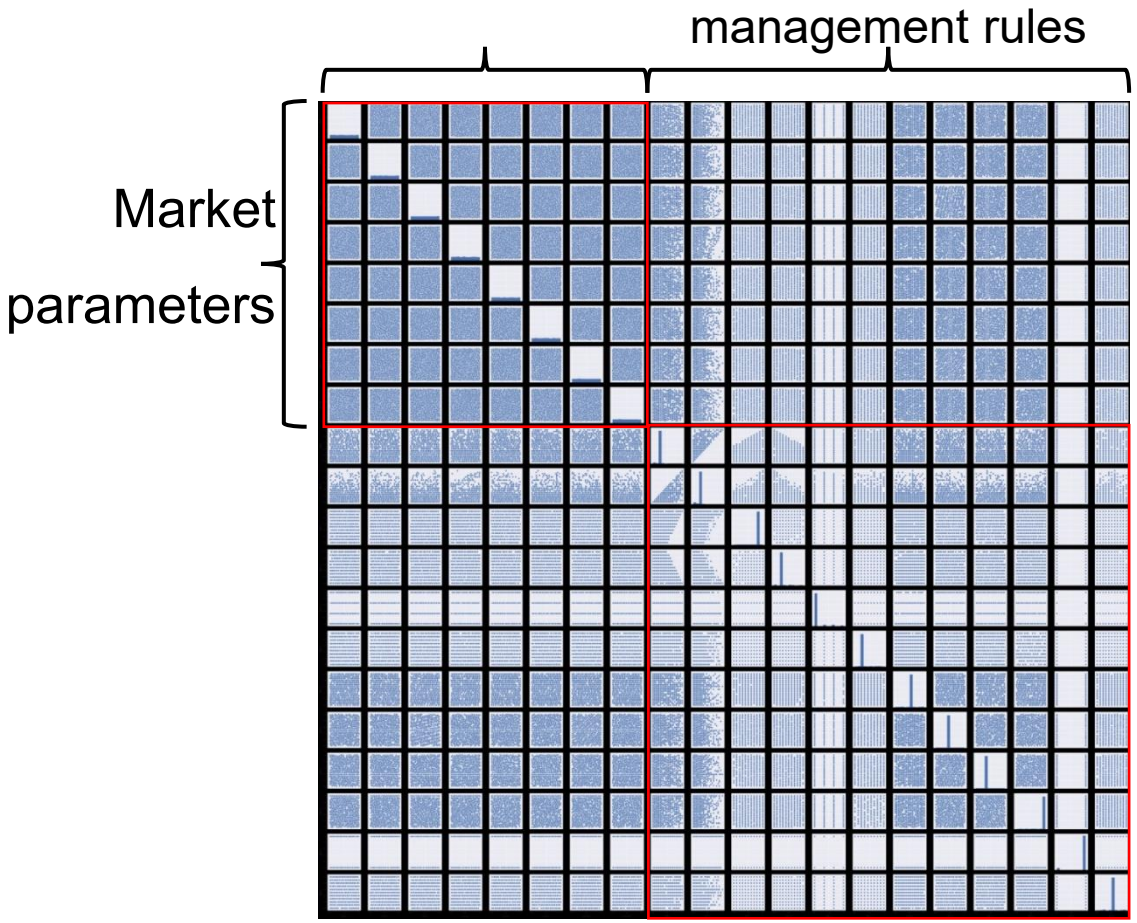
3 main outputs
- Aggregation of stochastic runs
  → Risk capital, solvency ratio

* Economic Scenario Generator

10’230 x 3 outputs

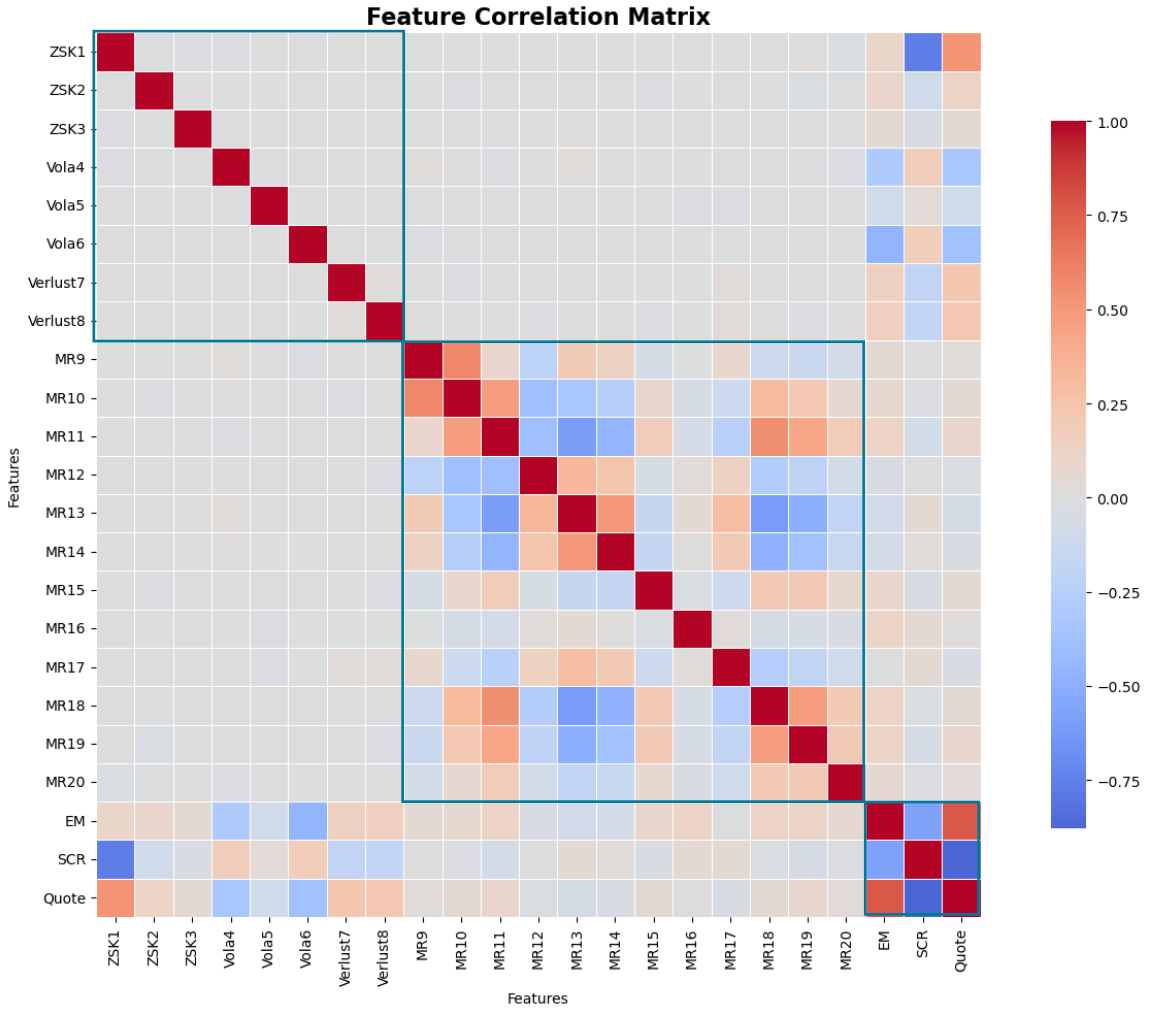


Market parameters in data set:
Covering the full [0,1] intervals and no correlation.



Pairplot: Distribution of input values

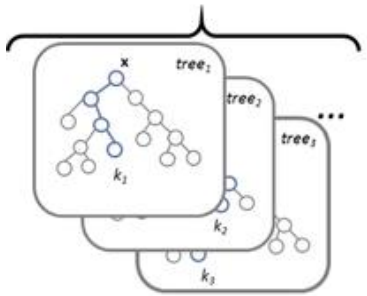
Correlations

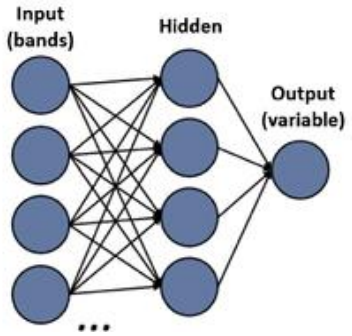


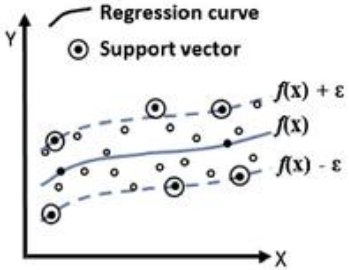
We explored a large number of model families and setups

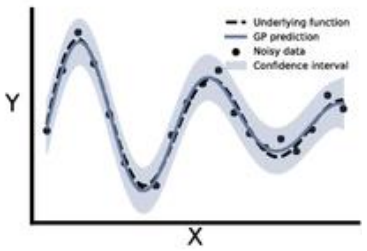
Model families:

- **Regression**: various linear models plus quadratic and cubic
- **Tree-based** models: Random Forest and XGBoost
- **Neural networks**







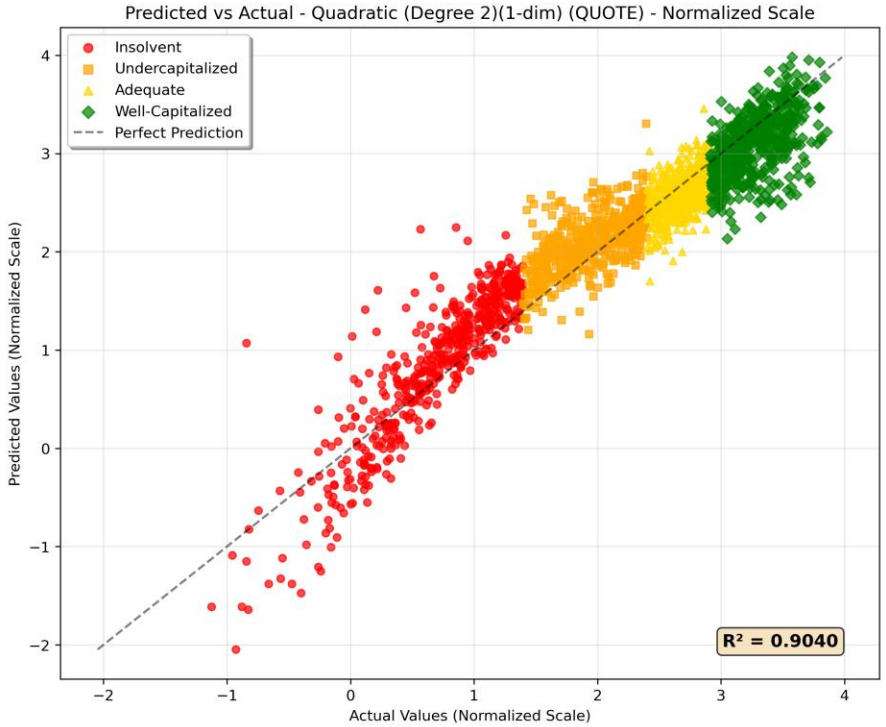


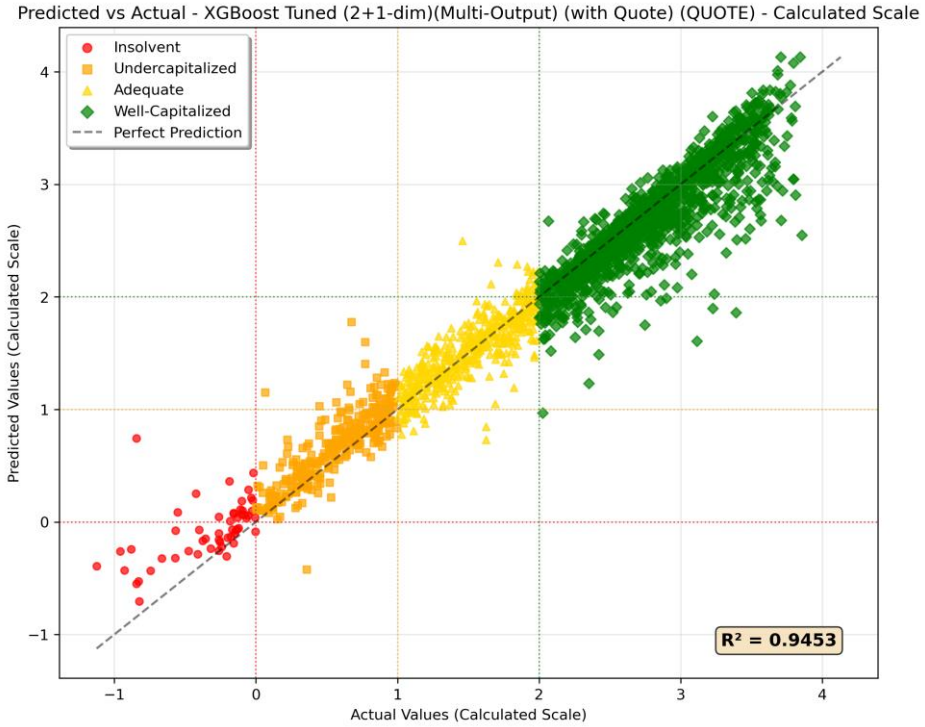
- **Grid search** for architecture / hyperparameters

- **Direct prediction** of solvency ratio vs. **indirect prediction** (= available / required capital)
  - Regression performed better on direct prediction
  - Indirect preferred as helps understanding of drivers for solvency ratio

XGBoost and **neural networks** outperformed other model classes

Predicted vs. actual plots for solvency ratio (perfect fit on line of identity)

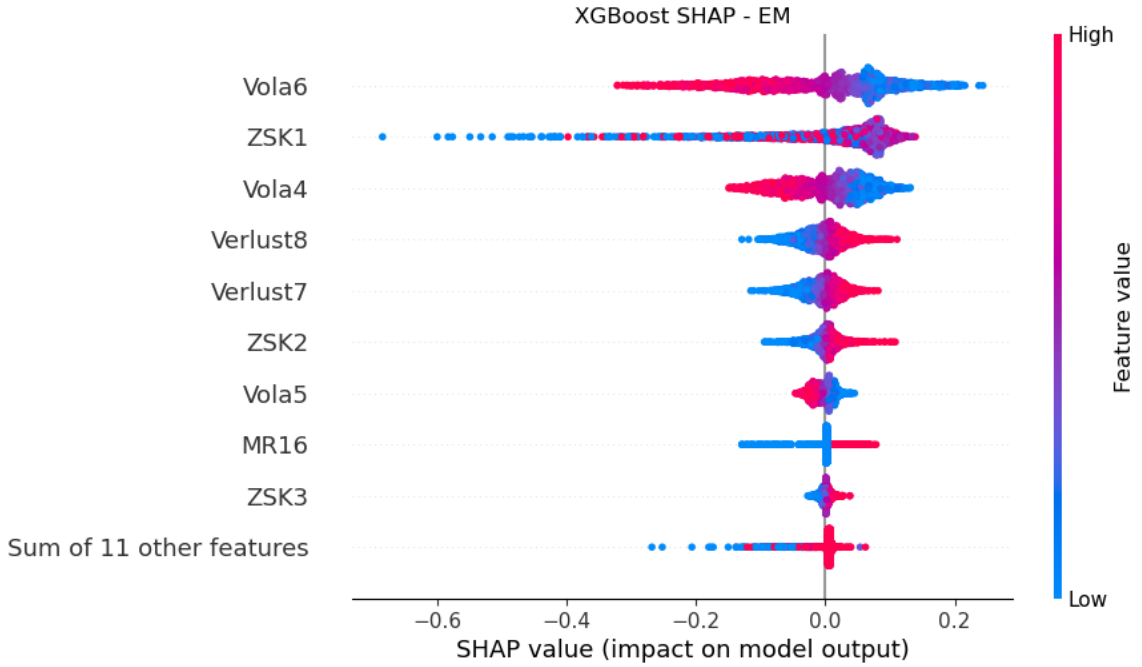




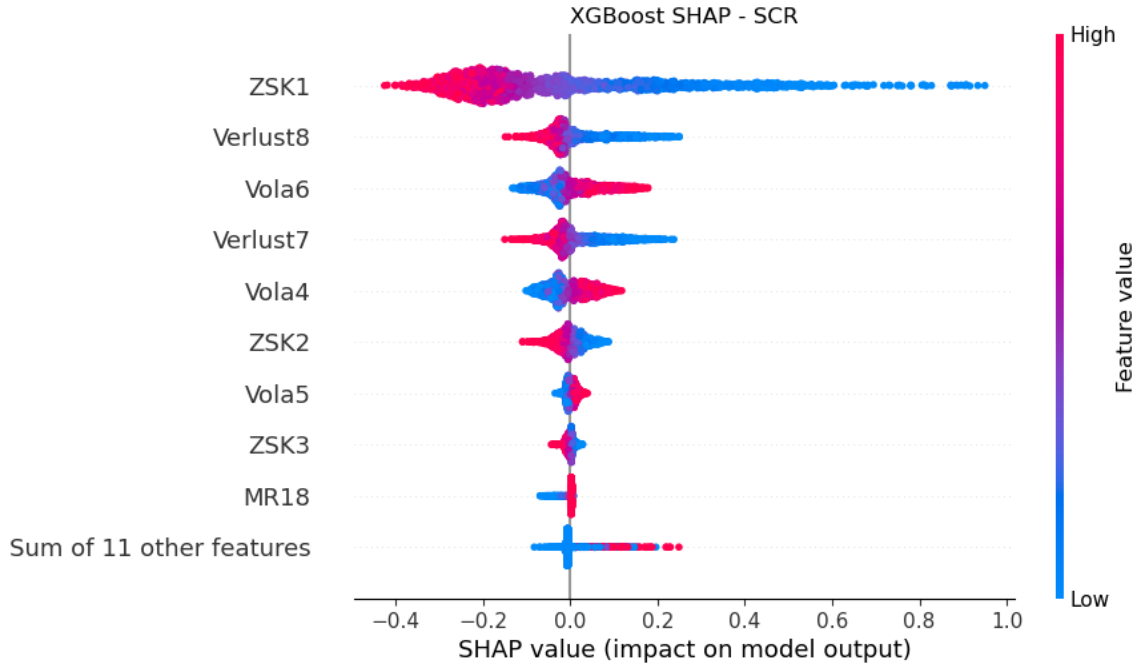
Best regression model (quadratic) vs. Gradient Boosted Trees

Explainability through SHAP values

- ZSK1 (≈ **interest rate level**) by far the most important feature
  → In line with actuarial intuition



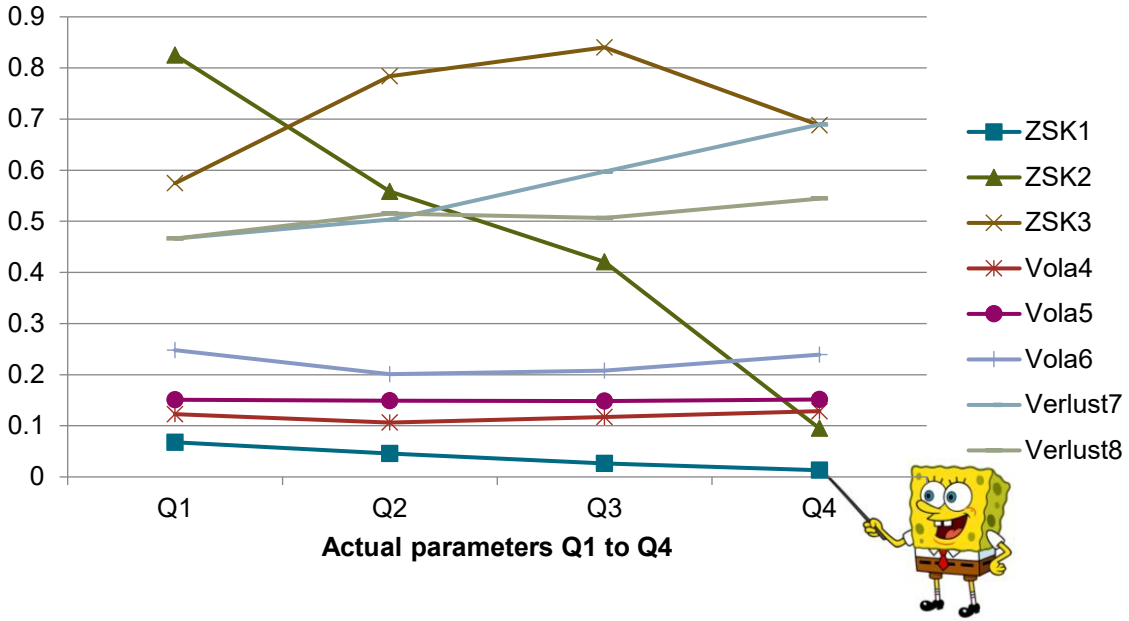
Beeswarm for available capital



Beeswarm for required capital

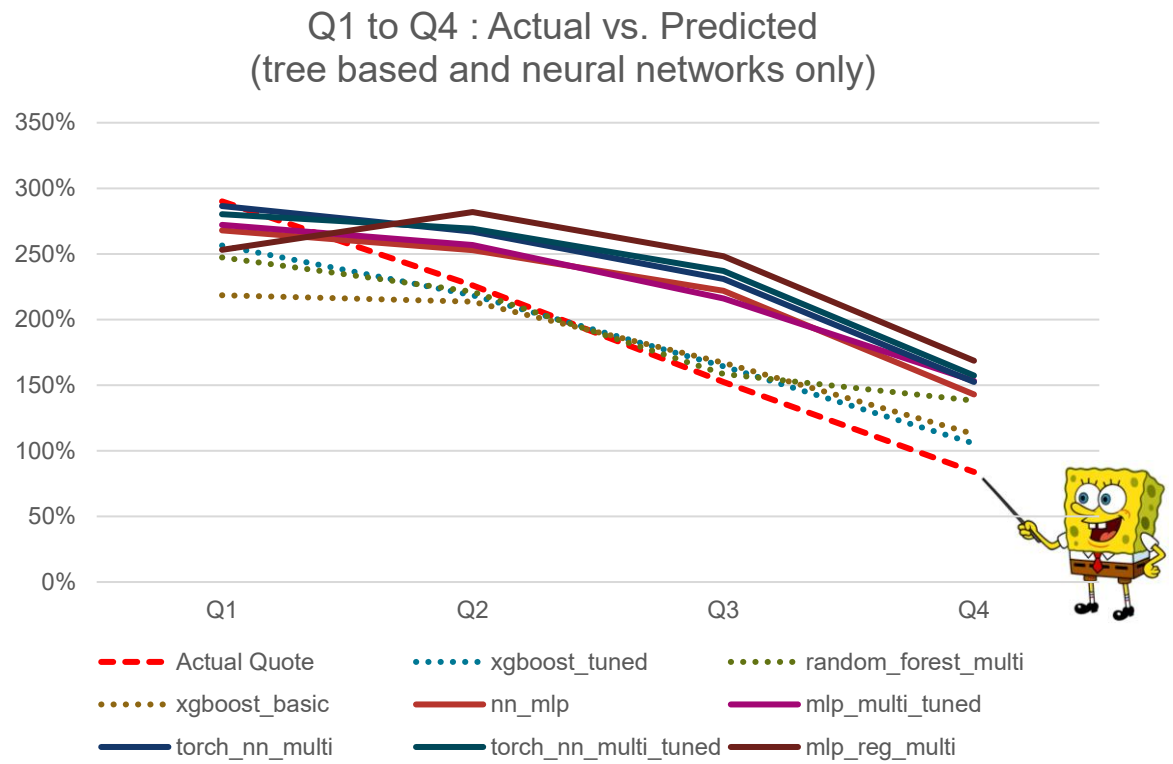
Back testing: XGBoost predicted observed change better than NN

Input parameters



ML models

Solvency ratio

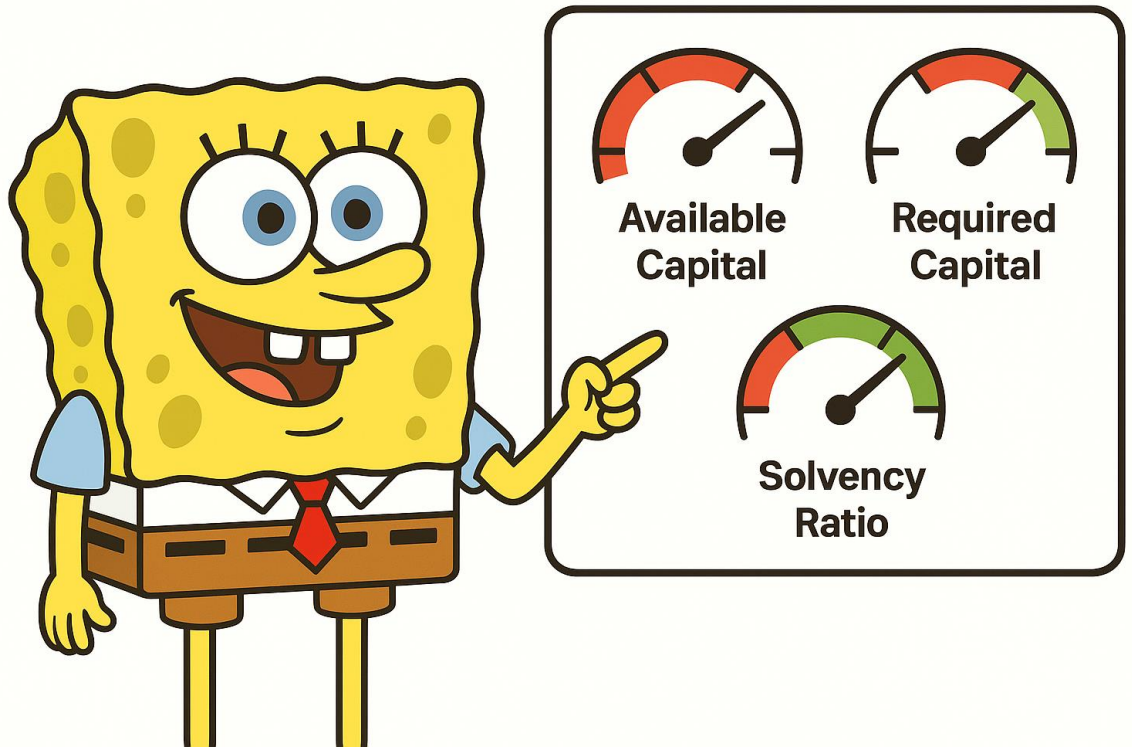


- ZSK1 moved from 0.07 to 0.01
  - SHAP: Available capital: ↓, required capital: ↑
- Ratio was dragged down from 290% to 84%

Use case: Dashboard for solvency ratio movement

Components:

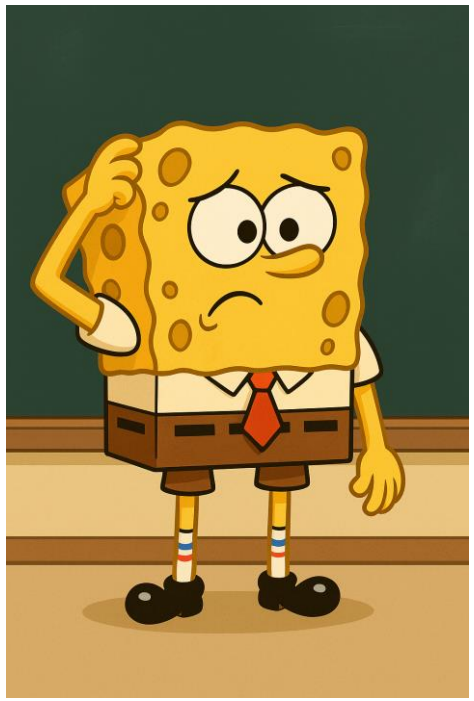
- Estimate available capital
- Estimate required capital
- Estimate solvency ratio
- Monitor:
  - Δ Economic inputs (fast-moving)
  - Δ Asset allocation (potentially fast-moving)
  - Δ Insurance business composition (slow-moving)



- Caveats:
  - Only indicator, not regulatory metric
  - Stability of model must be verified over multiple reporting periods (more research needed)

Many thanks to the two master students who supported me:

- Santiago Brunner
- Mahbod Tajdini



Time for Q&A (if there is time left…)

Extra points if you can recognise to which album cover the guitar refers to…