Evaluation von Modellen

„ Do machine learning like the great engineer you are, not like the great machine learning expert you aren’t.“

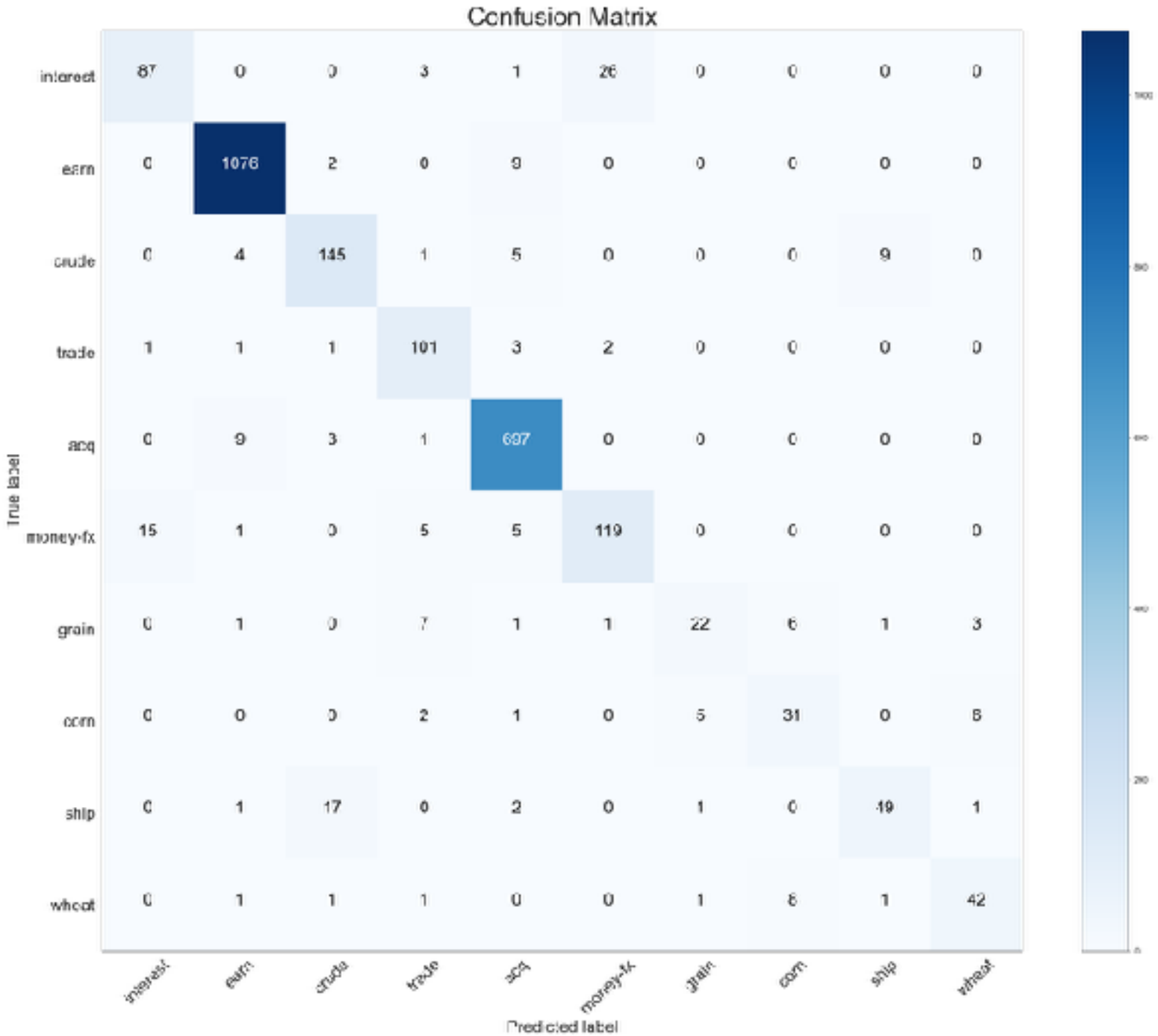
–Google Machine Learning Guide

Metriken / Verlustfunktionen

Metriken zur Evaluation

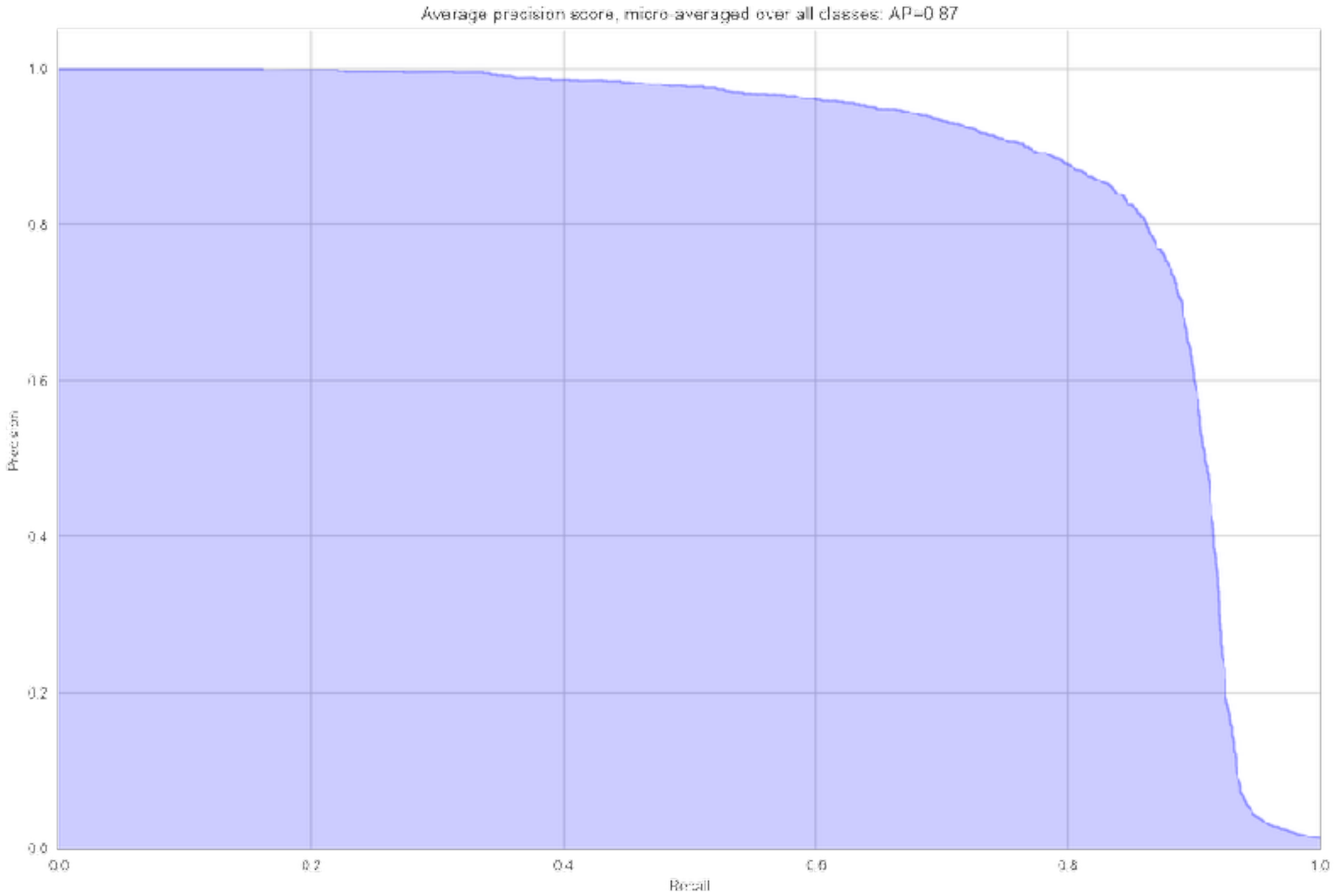
Sind ein Proxy für das, was man eigentlich verbessern möchte, aber das kann man oft nicht messen.

- Accuracy
- Precision / Recall / F1
- ROC / AUC
- RMSE / MAE
- MAP / NDCG / ERR



Confusion Matrix

True vs Predicted



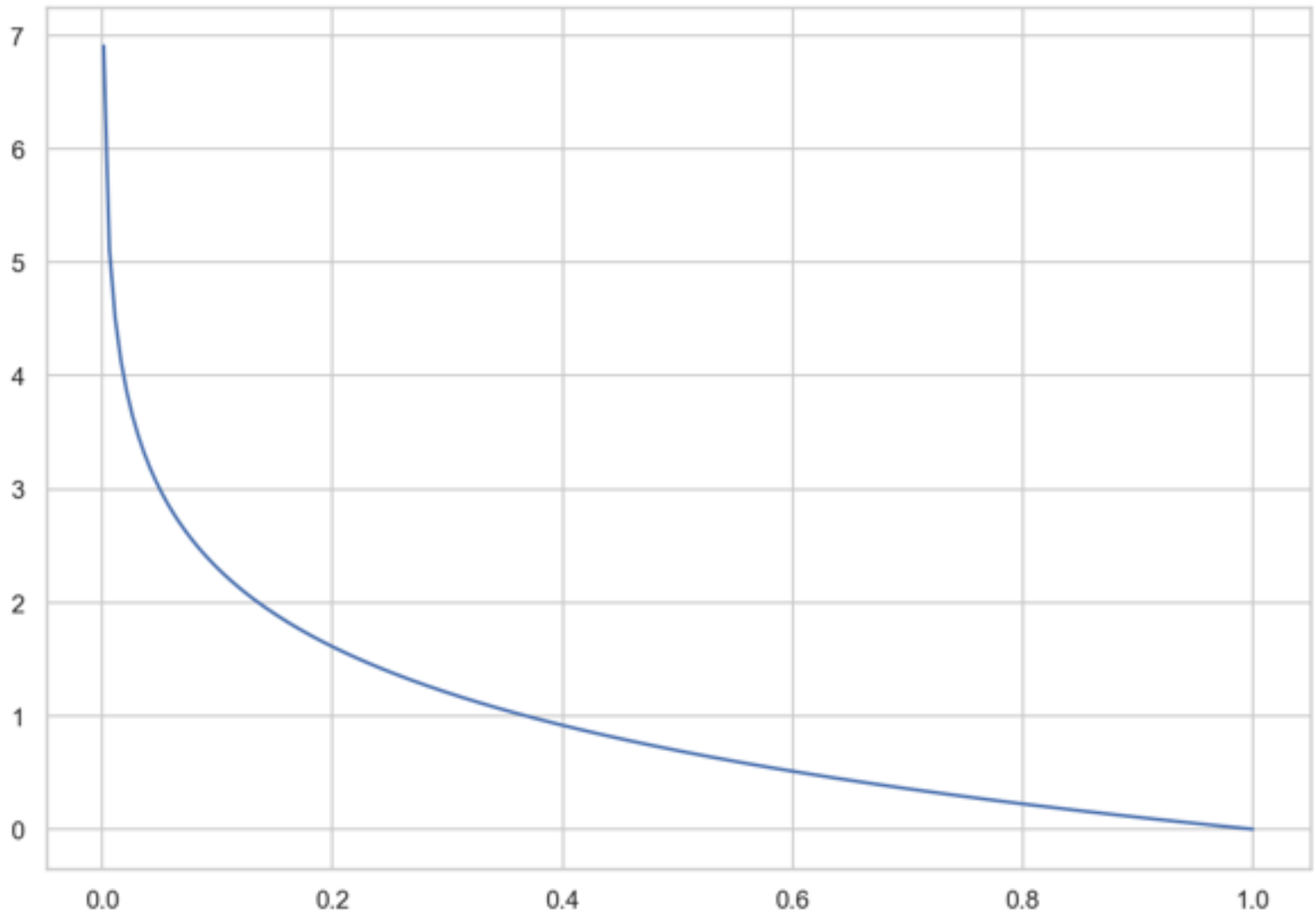
Precision / Recall

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

Verlustfunktionen sind das, was das Modell optimiert

Sind ein Proxy für die Metrik, die man optimieren möchte, aber die lässt sich oft nicht direkt als Verlustfunktion verwenden

- Logloss / Cross entropy
- Hinge loss
- 0-1 loss



Logloss

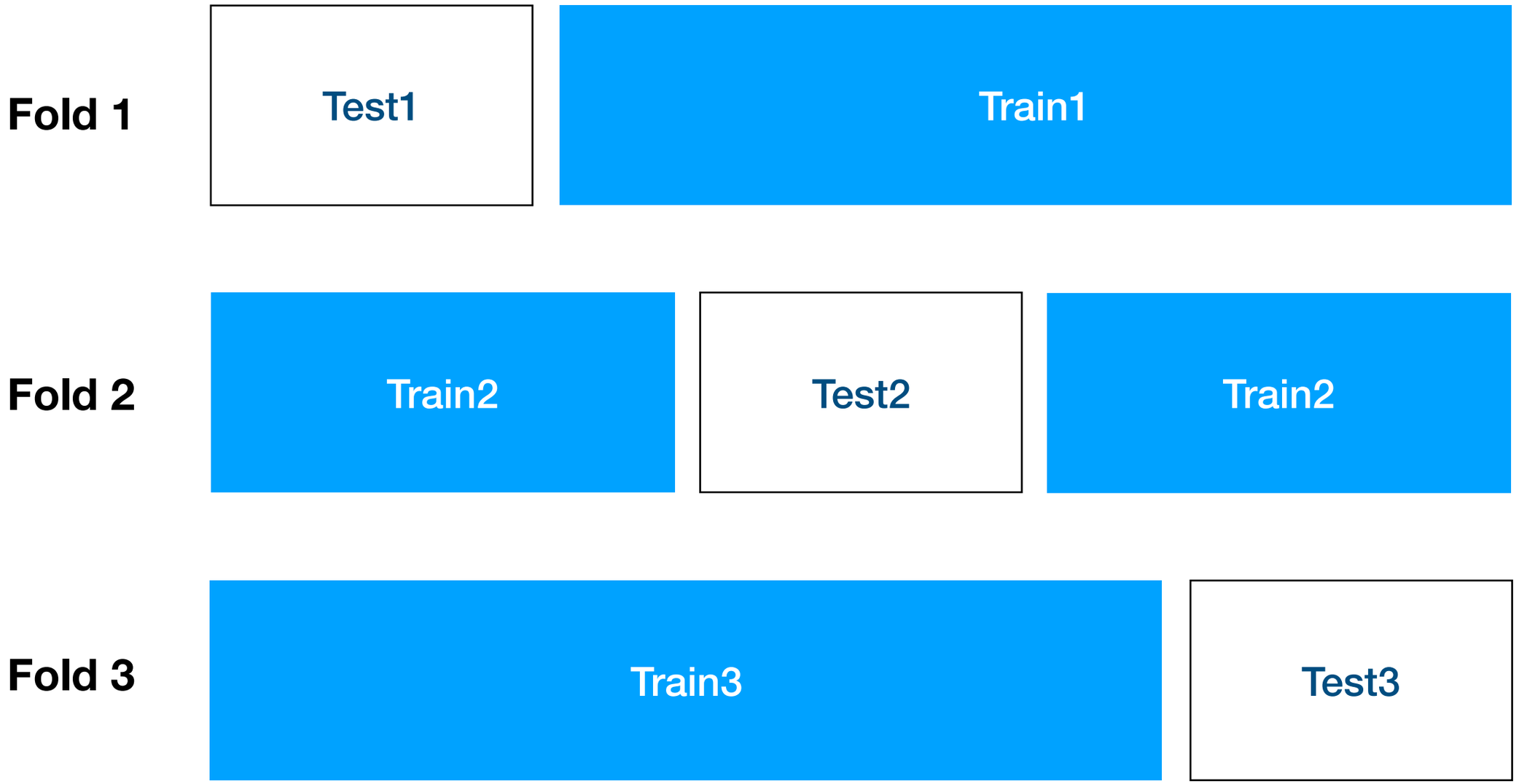
$$L_{\log}(y, p) = -\log \Pr(y \mid p) = -(y \log(p) + (1 - y)\log(1 - p))$$

Wie bekomme ich einen Score für mein Modell?

Aufteilen der der Trainingsdaten

- Training - wird zum Fitten des Models benutzt
- Validation - wird genutzt, um Änderungen am Modell zu bewerten
- Test - wenn ein Modell gut aussieht, lässt man es ein letztes Mal über das Testset laufen

K-fold Crossvalidation



Wichtige Punkte

- Man sollte die Folds so wählen, dass die Targets immer ungefähr gleich repräsentiert sind (StratifiedKFold)

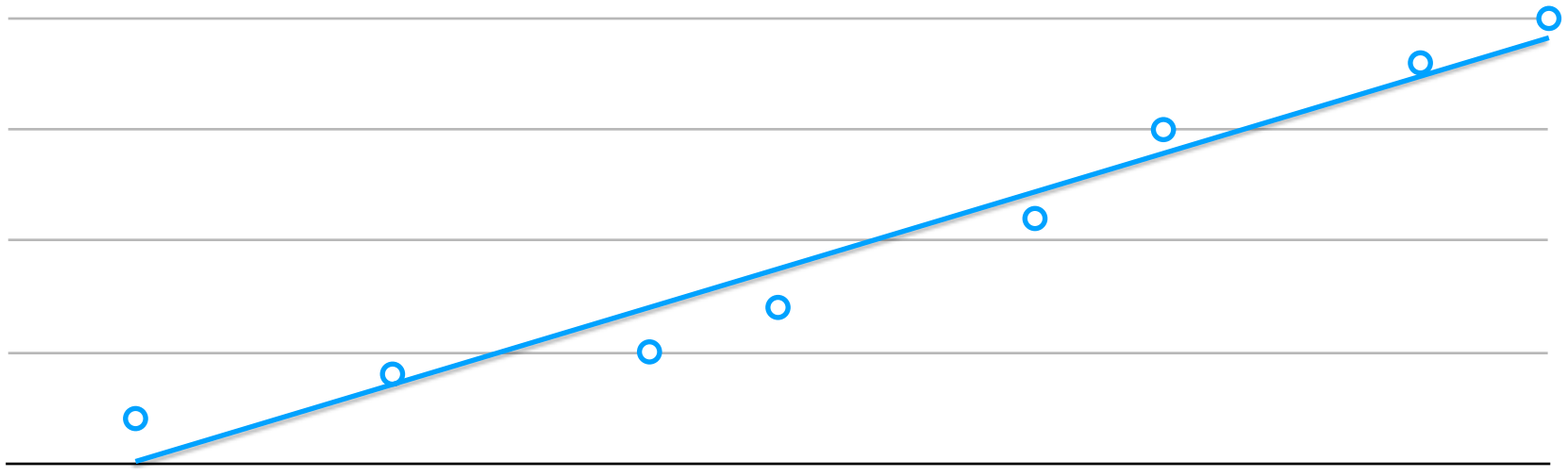
- Wenn man eine Zeitachse hat und die Zukunft vorhersagen möchte, sollten die Folds nach Zeit getrennt sein - ein Fold wäre dann beispielsweise ein Tag oder eine Woche

- Test und Trainingsdaten sollten sich nicht überschneiden

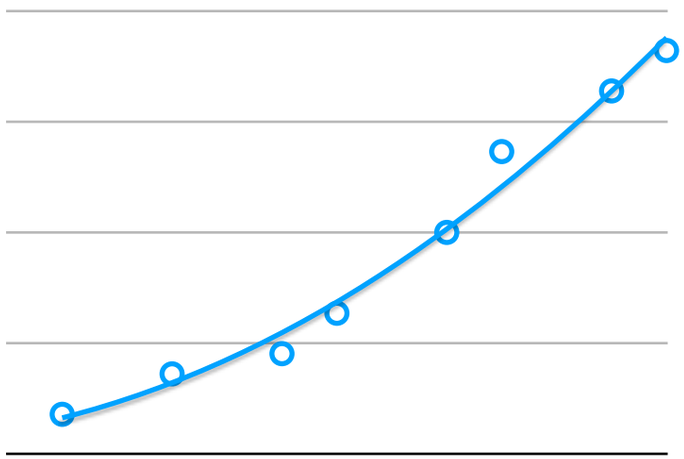
- Vielen Verfahren, die Daten zufällig teilen etc. kann man einen random_state=2018 mitgeben, damit scores vergleichbar bleiben, auch wenn man shuffled

Underfitting vs Overfitting

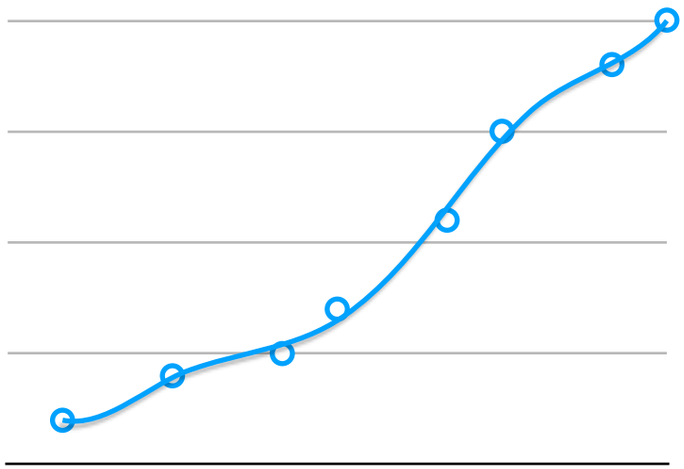
Linear



Quadratisch

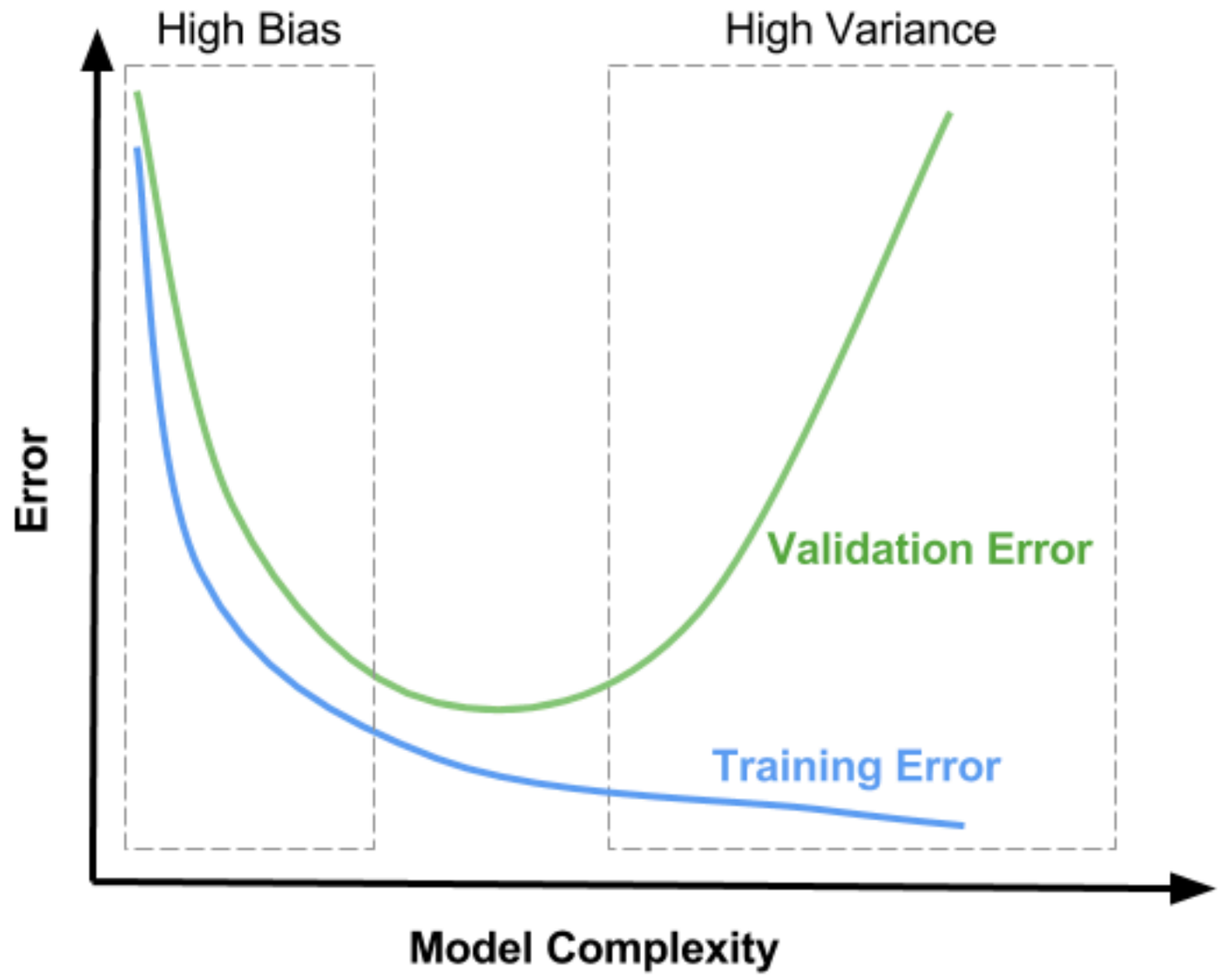


Polynom



Modelle unterschiedlicher Komplexität

Je komplexer das Modell, desto höher die Gefahr für Overfitting



Bias vs Variance

Fehlerquellen

- High Bias oder auch Underfitting - das Modell kann die Beziehung zwischen Input und Output nicht abbilden, lineares Modell für quadratische Daten beispielsweise

- High Variance oder auch Overfitting - Das Modell hat sich an die Trainingsdaten überangepasst und generalisiert nicht

- Irreducible Error - Rauschen in den Trainingsdaten

Regularisierung

Welche Arten von Regularisierung gibt es?

- L1 - Man erhöht den loss, wenn sich die Gewichte des Modells absolut erhöhen (L1-Norm)

- L2 - Man erhöht den loss, wenn sich das Quadrat der Modellgewichte erhöht (L2-Norm)

- Dropout, Lasso, Ridge, Early stopping

Wie baut man erfolgreiche Modelle?

- Erstellen einer verlässlichen Crossvalidation-Strategie, d.h. Änderungen im Score auf dem Validation-Set sagen voraus, wie sich der Score auf dem Test-Set verändern wird
- Wildes Feature-Engineering - dazu kommen wir noch..
- Testen von Modellen, die möglichst unterschiedlich sein sollten