MODELING CRYPTO TIME SERIES

A Machine Learning Approach to a Regression Problem

D. Giordano | D. Rasner | A. Redey

CAS Data Engineering - **Modul 2**

Davide

1. Daten Verstehen

6:00

Daten Quellen und Herausforderungen mit Zeitreihen

Akos

2. Vorhersage Modelle

+10:00

Übersicht Experimente

David

3. Ergebnisse Bewerten

+6:00

Vergleich und Bewertung von Ergebnissen

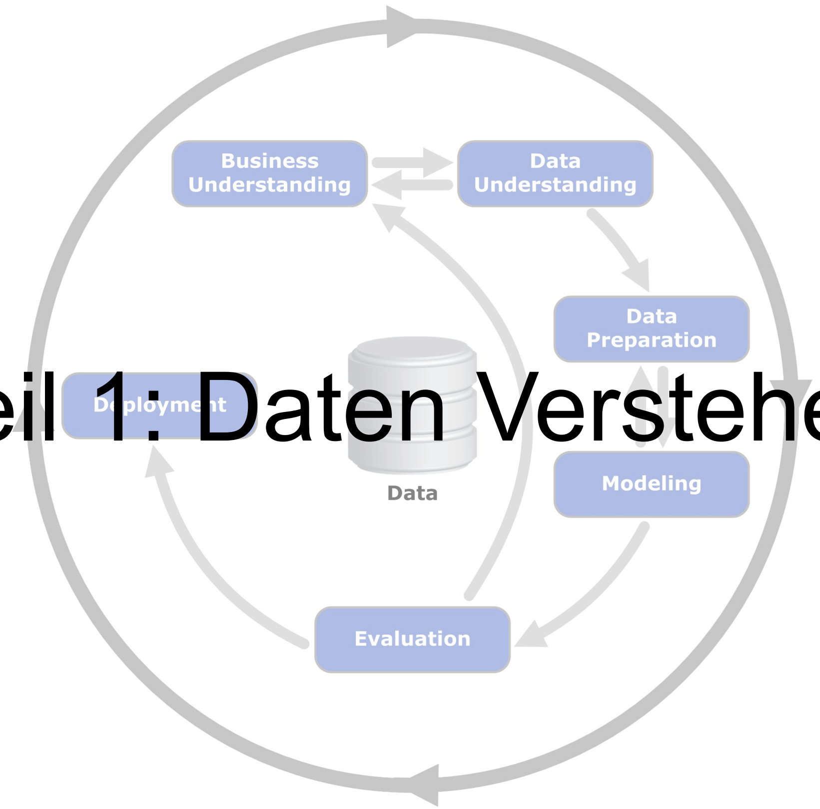
Alle

4. Lessons Learned

+6:00

Zusammenfassung...

Teil 1: Daten Verstehen



Teil 1: Daten Verstehen - **Zielsetzung**

Die Fragestellung:

Kann man historische Daten nutzen um ein Vorhersagemodell zu entwickeln?

- Wird der Preis für Bitcoin am nachfolgenden Tag steigen?
- Oder sinken?

Bild: BTC / USDT Preis, Candle Plot. Quelle: trandingview.com

Teil 1: Daten Verstehen - **Datenquellen und Feature Engineering**

Grundlage:

- Candlestick (Preis in USD pro Tag)
- Zeitrahmen: 2014 - 2021
- Fokus auf: Closing Preis (pro Tag)
- Target: **Closing Differenz (%) d-1**

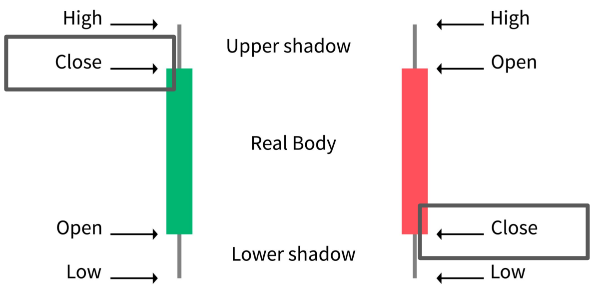


Bild: Candlestick. Quelle

Response:


```
[
  [
    1499040000000,      // Open time
    "0.01634790",       // Open
    "0.80000000",       // High
    "0.01575800",       // Low
    "0.01577100",       // Close
    "148976.11427815",  // Volume
    1499644799999,      // Close time
    "2434.19055334",    // Quote asset volume
    308,                // Number of trades
    "1756.87402397",    // Taker buy base asset volume
    "28.46694368",      // Taker buy quote asset volume
    "17928899.62484339" // Ignore.
  ]
]
```


Bild: Response JSON von Binance REST API

Teil 1: Daten Verstehen - **Datenquellen und Feature Engineering**

Feature Engineering:

- **Crypto Signals:** intotheblock.com
  - Financial signals
  - Network signals
- **Sentiment und Markt Aktivität:** lunarcrush.com
  - Reddit sentiment
  - Twitter sentiment
  - Telegram sentiment
  - Mentions and news score
- **Technicals:** Python Technical Analysis Library
  - Moving average und oscillators
  - Zu viele...
- **Aktien Preise:** alphavantage.co
  - Aktien und Index Preisentwicklung
- **DIY Scraping Twitter Sentiment**

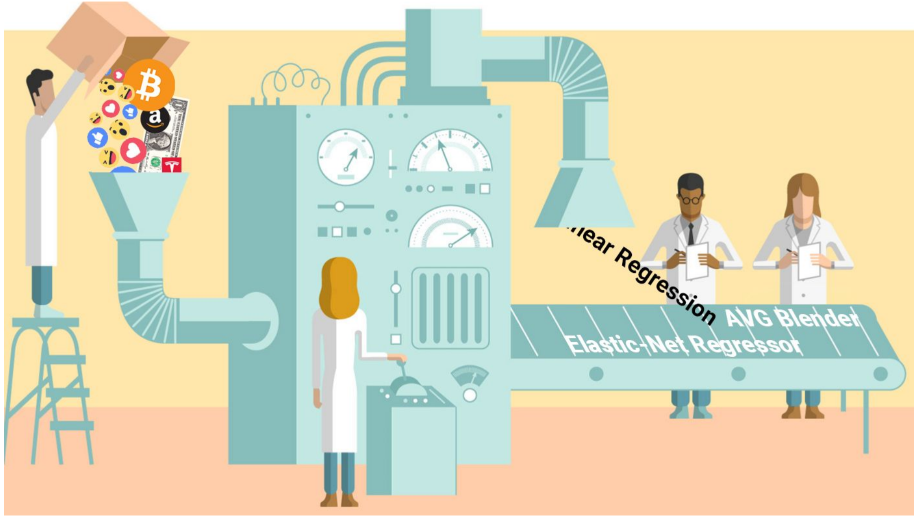


Bild: Unsere Feature Engineering Strategie

Teil 1: Daten Verstehen - **Herausforderungen mit Zeitreihen**

Problem 1: <u>Es gibt kein falsch...</u>

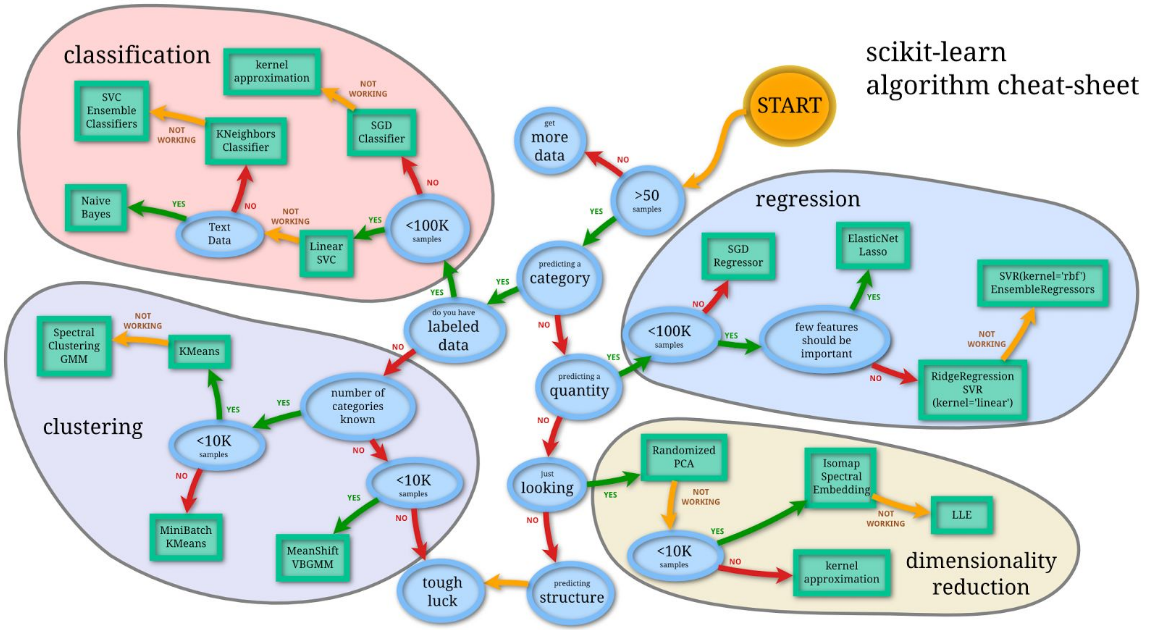


Bild: scikit-learn algorithm cheat-sheet. [Quelle]

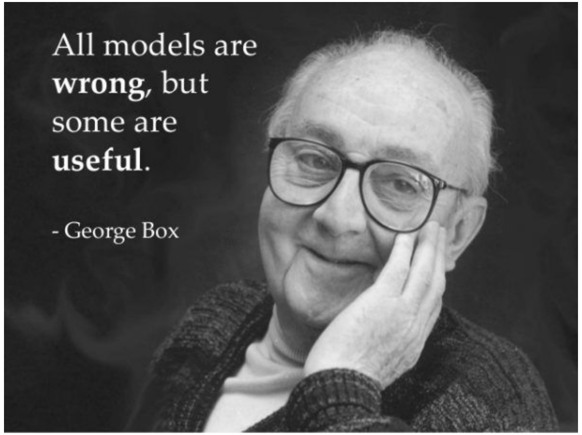


Bild: Zitat George Box. [Quelle]

Es gibt endlose Strategien und Alle könnten ein Ergebnis produzieren...

Teil 1: Daten Verstehen - **Herausforderungen mit Zeitreihen**

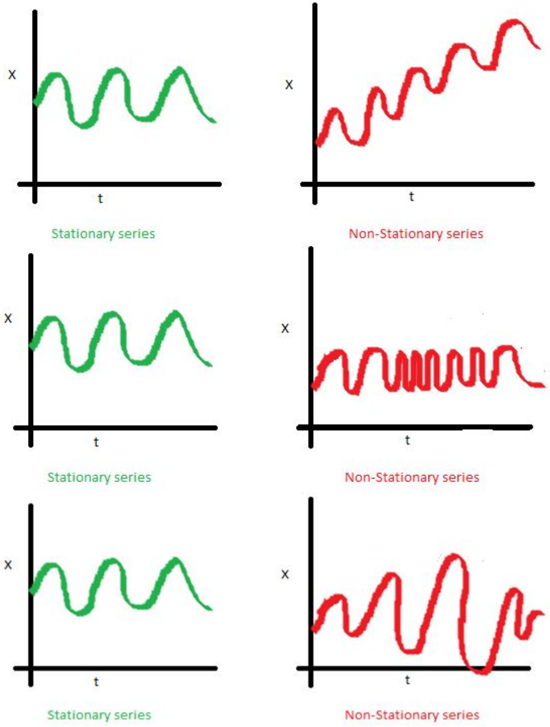
Problem 2: <u>Zeitreihen und Stationarität:</u>

Klassische TS Modelle gehen davon aus, dass die Datengrundlage **stationär** ist.

Frage:

➔ **Ist die Bitcoin Preis Entwicklung stationär?** Wenn nicht, wie geht man damit um?

Teil 2: Akos



Trend
(BTC adoption)
NO

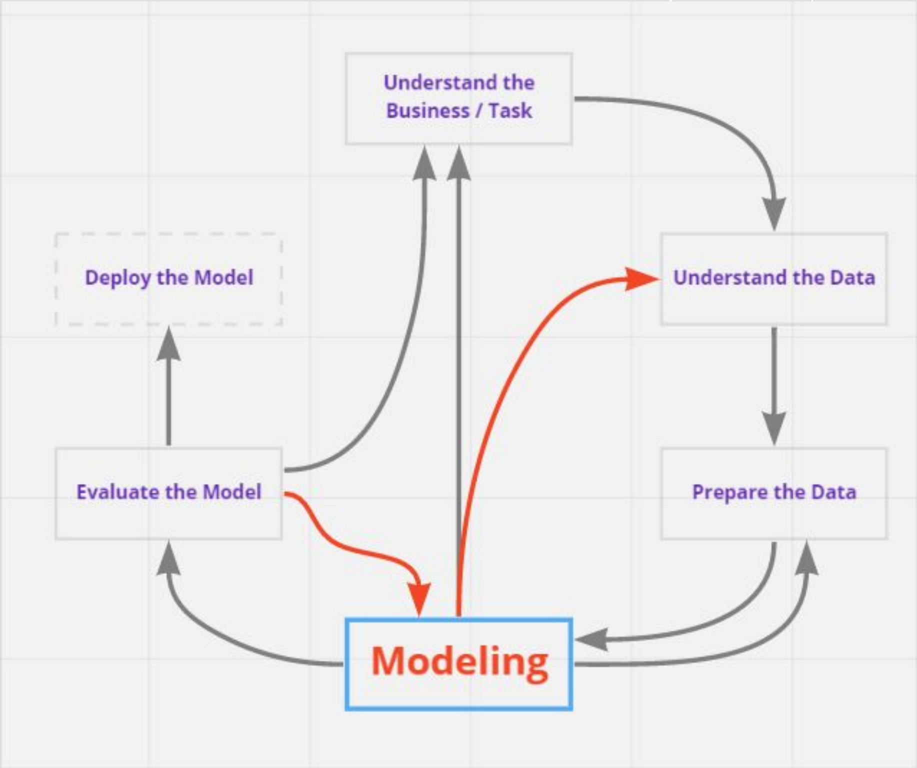
Seasonality
(BTC halving)
NO

Outliers
(BTC volatility)
NO

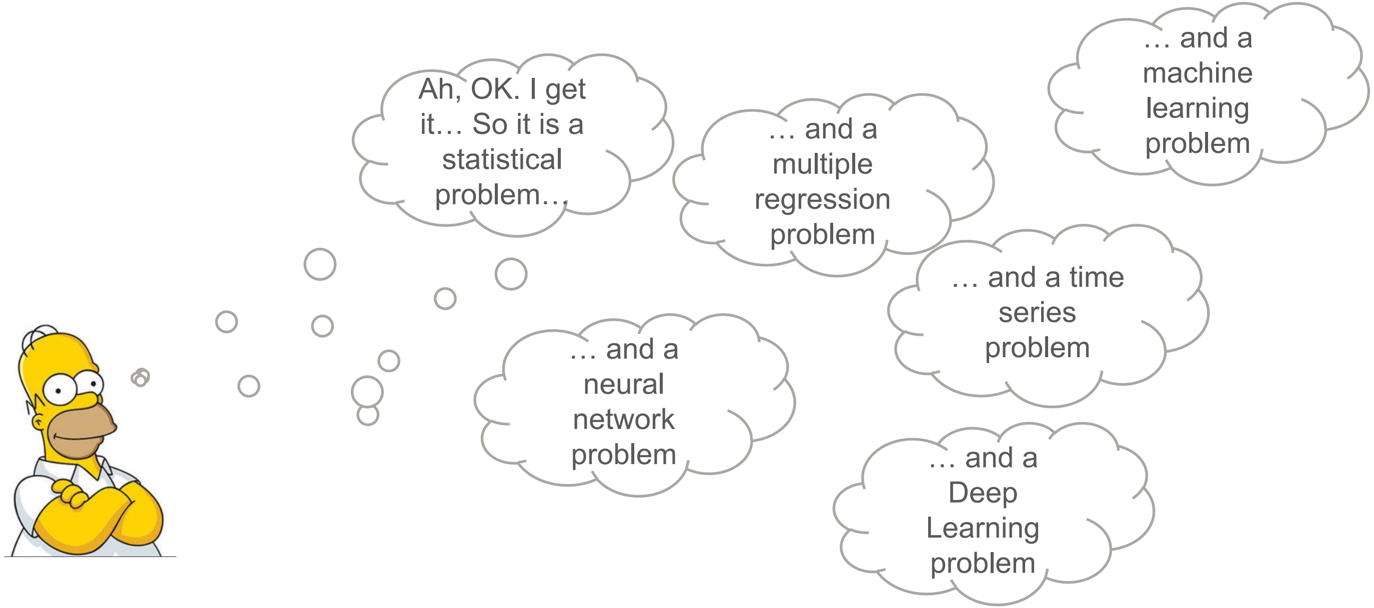
Bild: Stationäre und nichtstationäre Daten. [Quelle]

Teil 2: Vorhersagemodelle

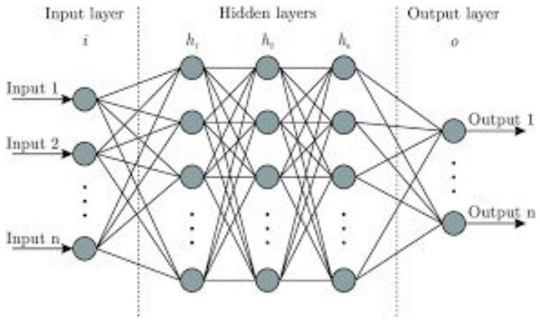
PROBLEM STATEME

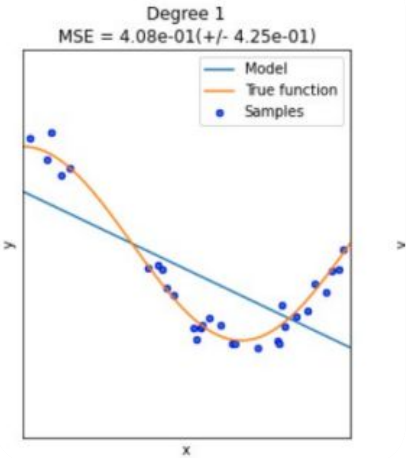


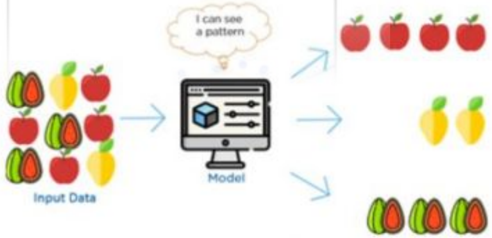
Teil 2: Vorhersagemodelle - **Herangehensweise**



Teil 2: Vorhersagemodelle - **Herausforderung**








```
arima_2020_tr <- auto.arima(bit_ts_tran2)

checkresiduals(arima_2020_tr)
```

```
fit_model = function(bitcoin_data, h){
  bitcoin_df = bitcoin_data %>%
    filter(Date >= as.Date('2017-01-01'))
    arrange(Date)

  time_series = bitcoin_df %>%
    select(WeightedPrice) %>%
    ts()

  predictions = time_series %>%
    BoxCox(lambda = BoxCox.lambda(time_ser
    auto.arima() %>%
    forecast(h)

  forecast_df = cbind(data.frame(predictio
                      data.frame(predictio
                      data.frame(predictio
```

Teil 2: Vorhersage Modelle - **Our initial thoughts**

OUR GOAL

Develop **prediction algorithms** for bitcoin prices, based on a time series dataset, consisting of financial, blockchain-related, technical analysis and sentiment daily signals.

Teil 2: Vorhersagemodelle - **Statistik: Vorgehensweise**

STATISTICAL TESTS

Test the assumptions of multiple linear regression and the requirements for modeling

FEATURE SELECTION

Check the correlation between predictors and their justification to be included

BUILDING & TUNING MODELS

Split the dataset into training and validation sets, train your models, perform cross-validation, tune the hyperparameters and select the best performing estimators.

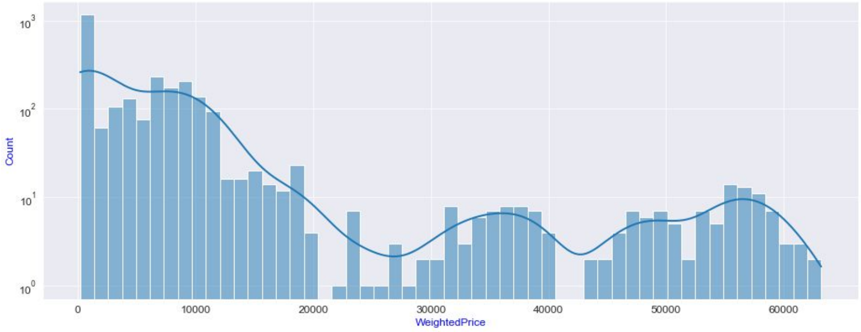
EVALUATION

Based on the selected and tuned models, make a prediction on the test set, measure the performance.

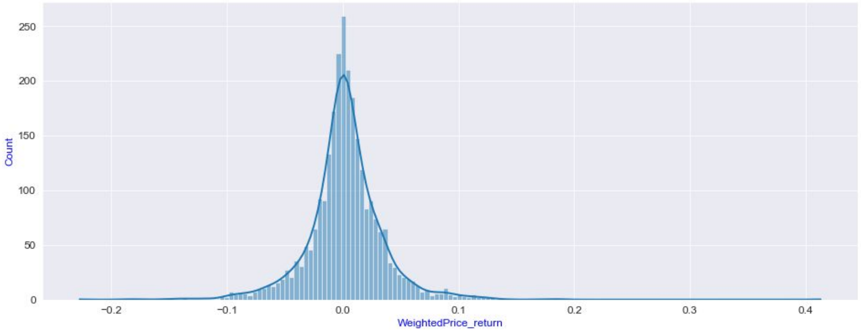
Teil 2: Vorhersagemodelle - **Statistik**

Teil 2: Vorhersagemodelle - **Statistik: Verteilung**

ORIGINAL

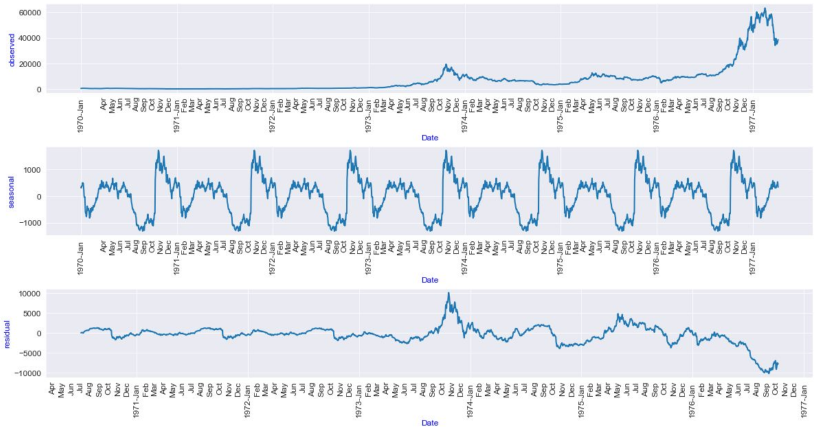


TRANSFORMED



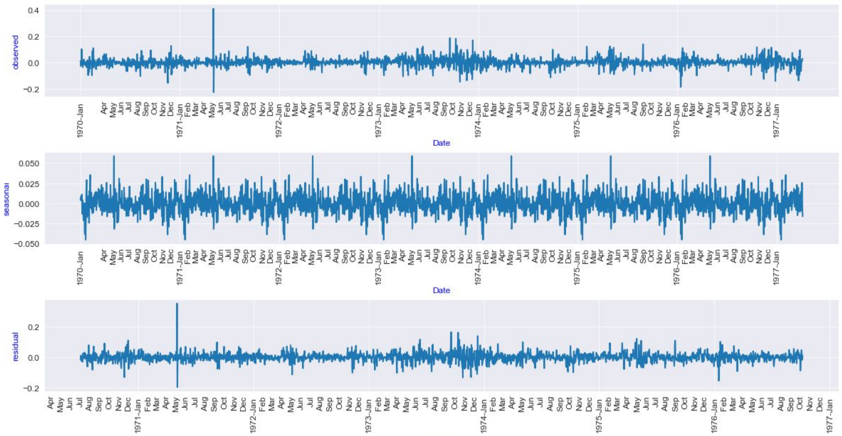
Teil 2: Vorhersagemodelle - **Statistik: Stationarität**

DECOMPOSITION ORIGINAL




```
ADF Statistic: -0.854419651726885
n_lags: 0.802612041825002
p-value: 0.802612041825002
```

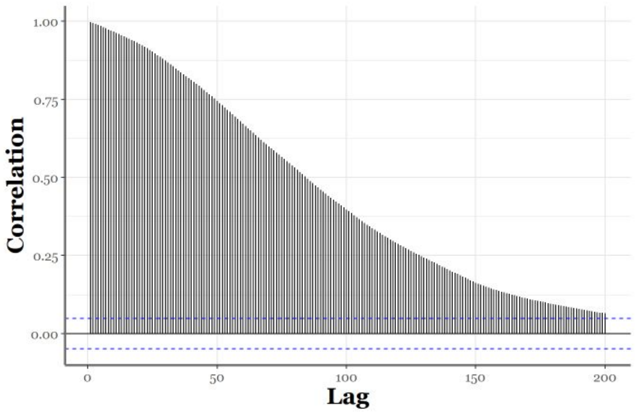
DECOMPOSITION TRANSFORMED



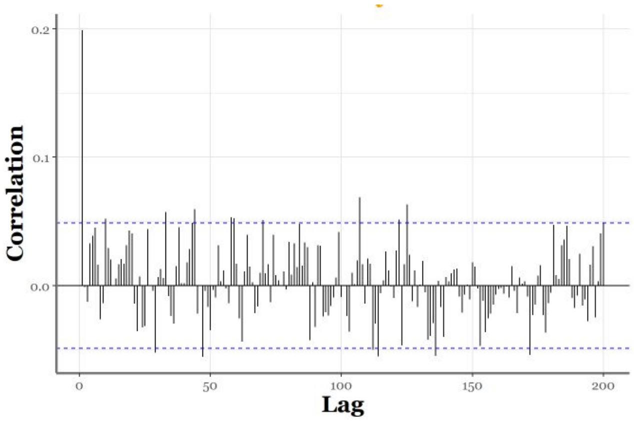

```
ADF Statistic: -34.97043776259842
n_lags: 0.0
p-value: 0.0
```

Teil 2: Vorhersagemodelle - **Statistik: Autokorrelation**

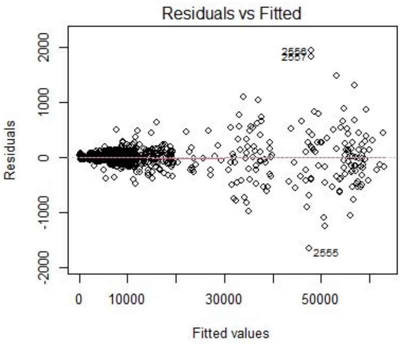
ACF ORIGINAL

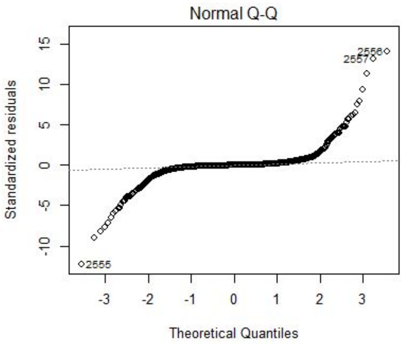


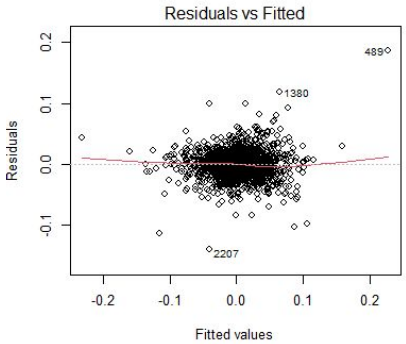
ACF TRANSFORMED

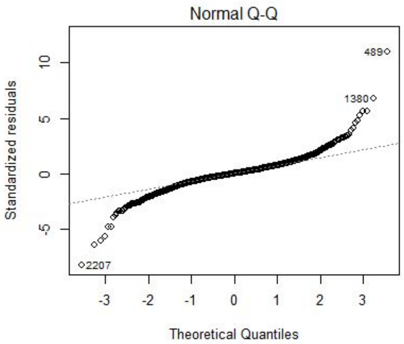


Teil 2: Vorhersagemodelle - **Statistik: Heterodeskedacity**










```
> # Breusch Pagan Test
> lmtest::bptest(lmMod)

	studentized Breusch-Pagan test

data:  lmMod
BP = 975.99, df = 46, p-value < 2.2e-16

>
> # NCV Test
> car::ncvTest(lmMod)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 12287.19, Df = 1, p = < 2.22e-16
```

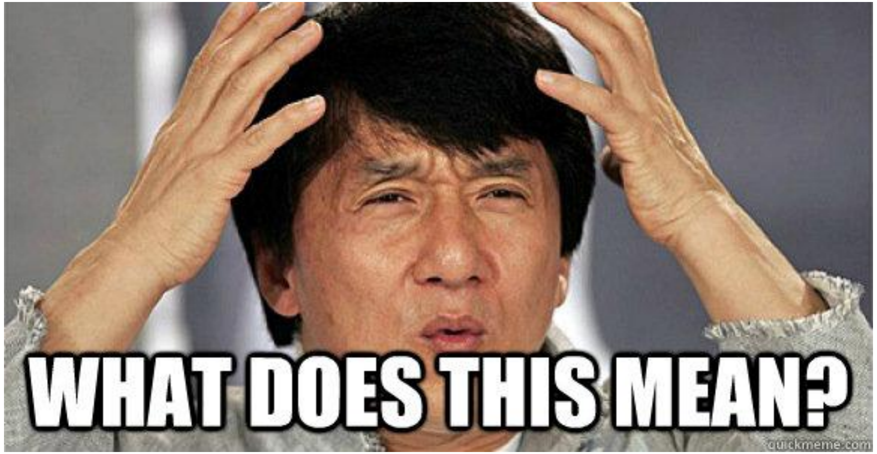
```
> # Breusch Pagan Test
> lmtest::bptest(lmMod_ret)

	studentized Breusch-Pagan test

data:  lmMod_ret
BP = 488.71, df = 46, p-value < 2.2e-16

> # NCV Test
> car::ncvTest(lmMod_ret)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 419.7224, Df = 1, p = < 2.22e-16
```

Teil 2: Vorhersagemodelle - **Statistik: OK?**



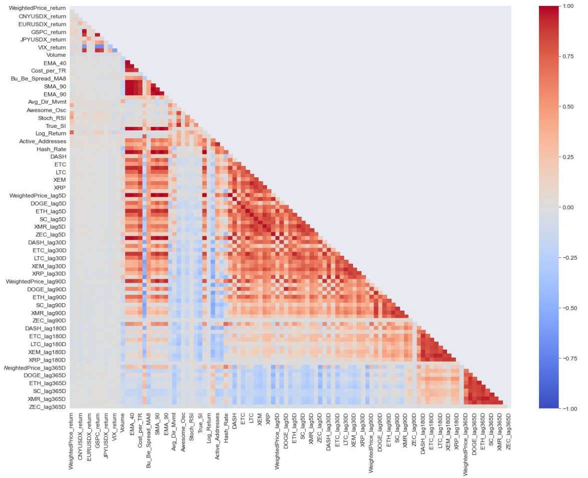
Teil 2: Vorhersagemodelle - **Statistik: Ergebnis**

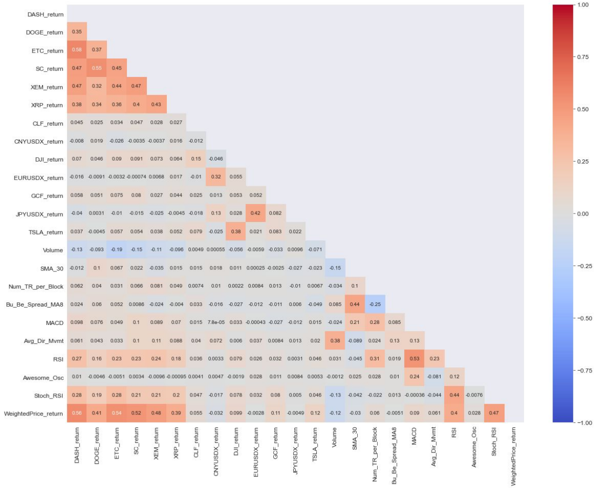
Test	Outcome	Meaning
ADF Stationarity test	Daily returns stationary	Linear Models can be used
ACF Plot (Autocorrelation)	Autocorrelation	Possible use case for ARIMA*
Breusch-Pagan test (Homoskedasticity)	Heteroskedasticity present: Errors vary	Possible use case for volatility clustering (GARCH)*

* ARIMA and GARCH models were not in scope of the project, although we experimented them in R a possible forecast with ARIMA in Appendix

Teil 2: Vorhersagemodelle - **Feature Selection**

Teil 2: Vorhersagemodelle - **Correlation Matrix**

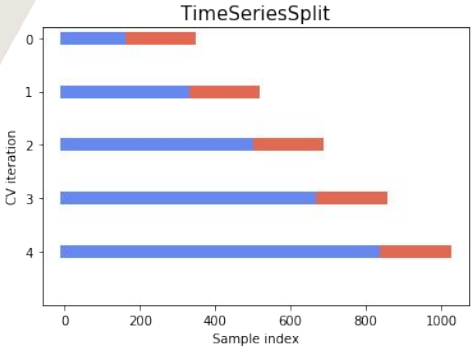




Teil 2: Vorhersagemodelle - **Model Building & Tuning**

Teil 2: Vorhersagemodelle - **Model Building & Tuning**

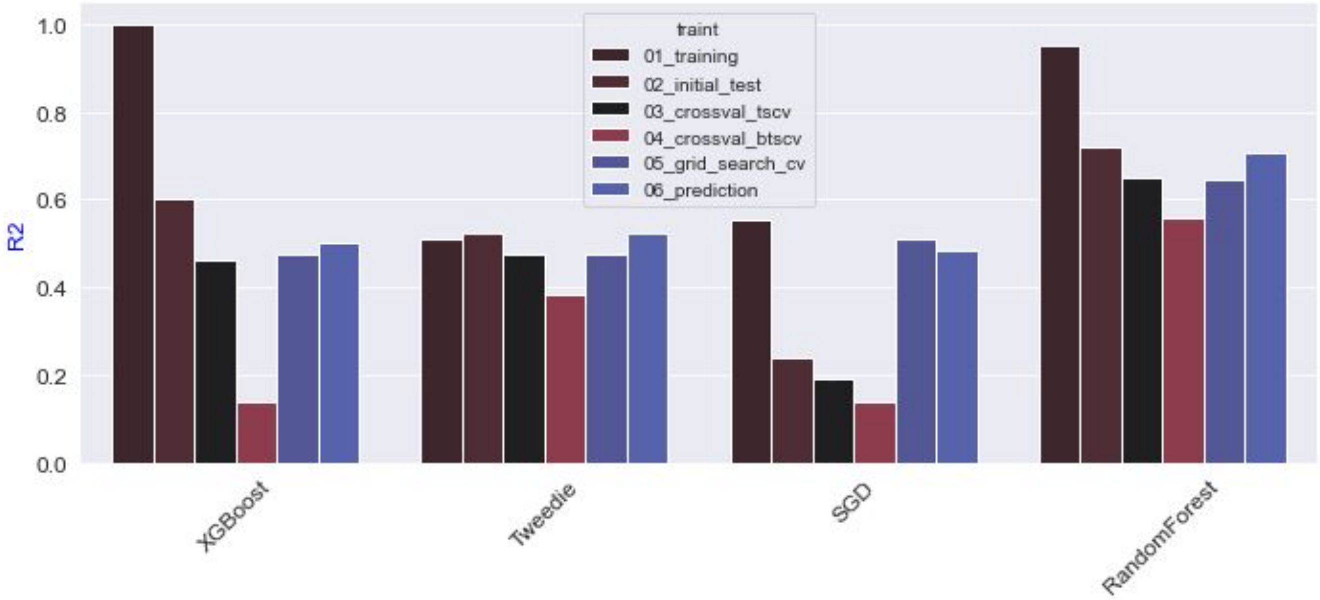
FOCUS: CROSS-VALIDATION & GRID SEARCH



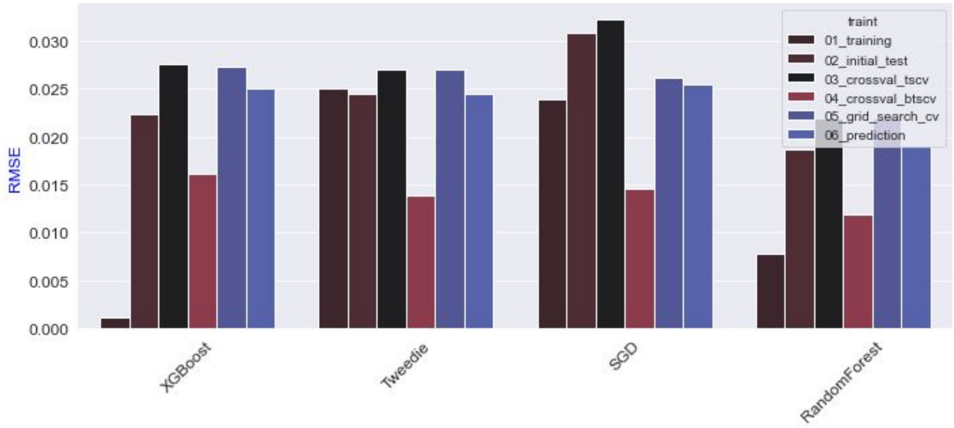
- Train/Test Split not shuffled
- Test Data saved for prediction
- Training – > Cross Validation with base models
- Extract parameters, define grid around it –> perform grid search with cross validation
- Select best estimators
- Save model artifacts
- Load models and perform prediction on the separated test dataset

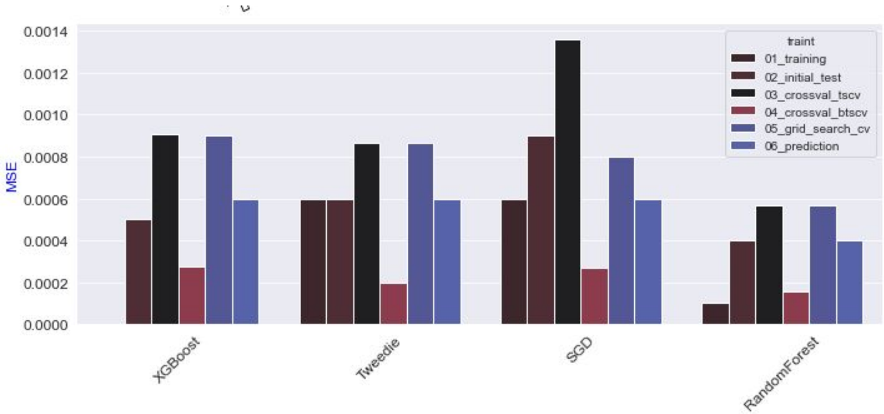
Teil 2: Vorhersagemodelle - **Evaluation**

Teil 2: Vorhersagemodelle - **Metrics: Model Fit**



Teil 2: Vorhersagemodelle - **Metrics: Error**





Teil 2: Vorhersagemodelle - **Zusammenfassung**

Our main focus was that we select the models, which show the best fit and tried to optimize their R-Squared.

We also checked that the errors either decrease or do not get worse.

Our 4 models, which stayed in scope of our optimization were the following:

➔ Tweedie Regressor
➔ Random Forest Regressor
➔ Stochastic Gradient Descent
➔ XGBoost Regressor

The most robust model in our analysis was the Tweedie Regressor, which is a Generalized Linear Model and is used to model data that follow Tweedie or Poisson distribution, which is the case in our project.

POSSIBLE EXTENSIONS

- Look for more / different features data (however data quality is a challenge)
- Get data with more granularity and frequency (eg. minutely, secondly)
- Try to approach the problem from a different angle (eg. Classification problem, Volatility clustering…)
- Transform the problem to a Deep learning project
- ….

Teil 3: Ergebnisse Bewerten

Teil 3: Ergebnisse Bewerten - **Lift Chart**

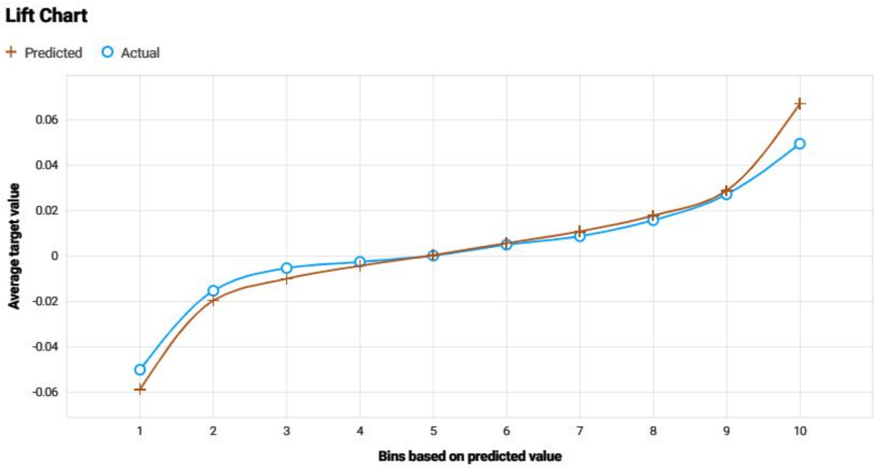


Bild: Lift Chart, Linear Regression (R Squared = 0.6955, -1.0445)

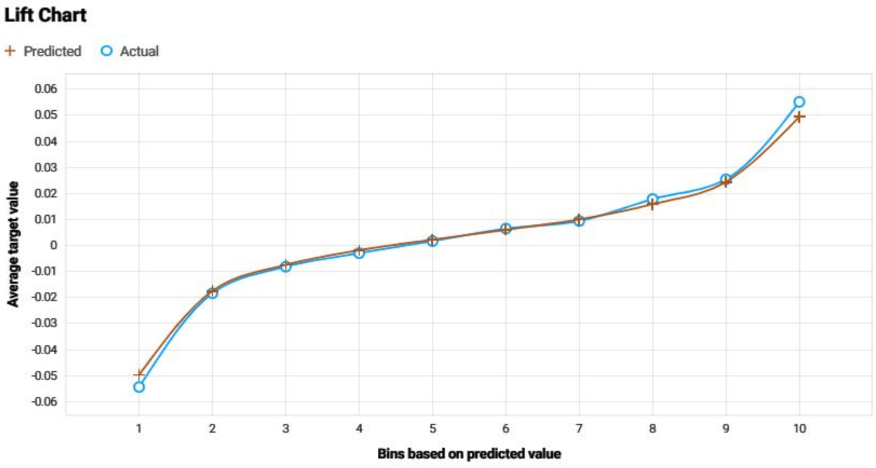


Bild: Lift Chart, Random Forest Regressor (R Squared = 0.7998, 0.7522)

Teil 3: Ergebnisse Bewerten - **Feature Importance**

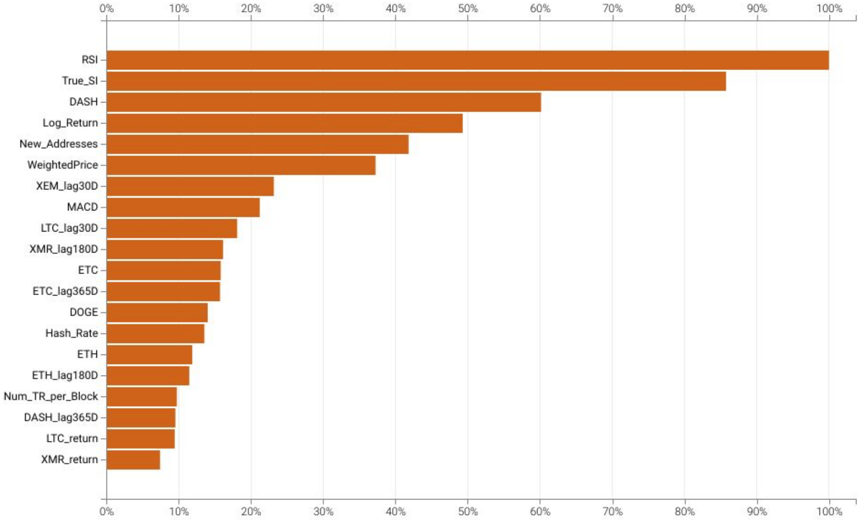


Bild: Feature Importance, Linear Regression

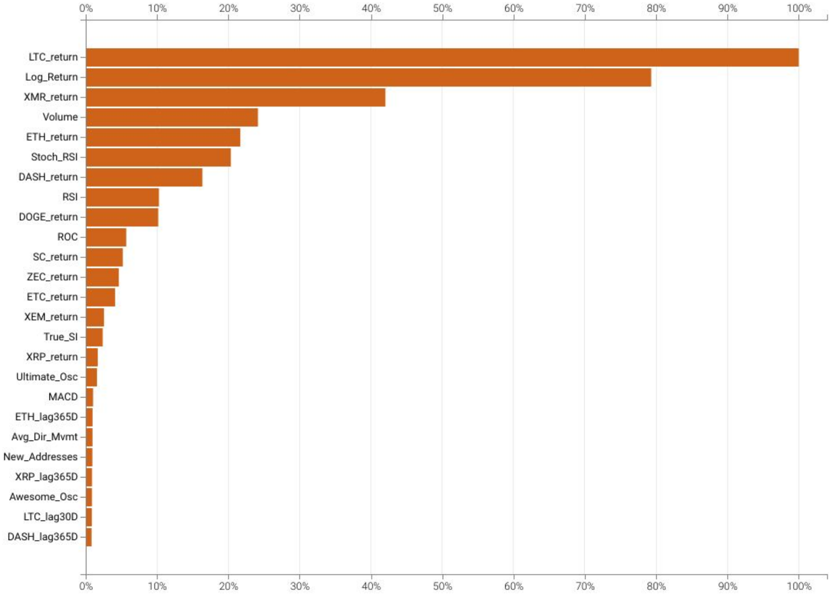
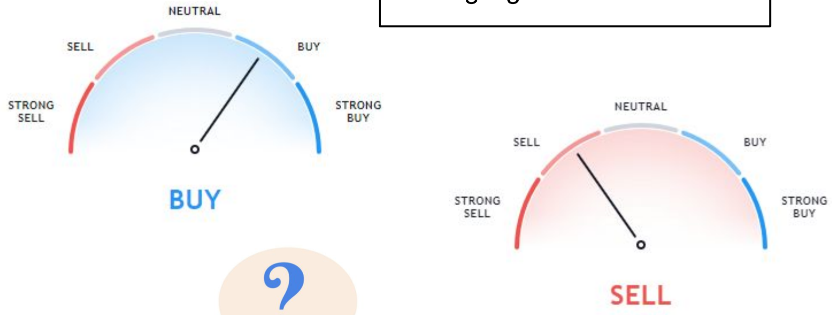
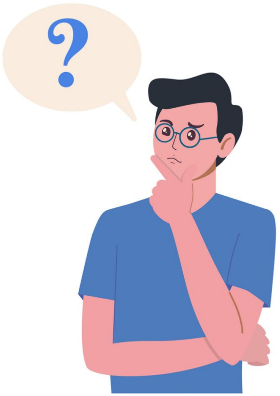


Bild: Feature Importance, Random Forest Regressor

Teil 3: Ergebnisse Bewerten - **Prediction Test**

Wie entsteht aus dem ein trading signal?



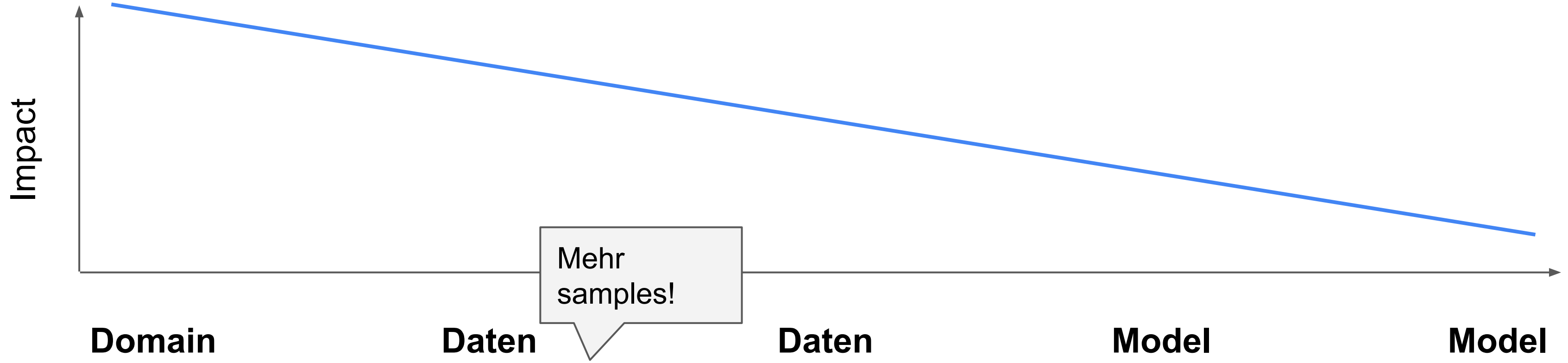


	A	B	C	D	E
1	row_id	Date	WeightedPrice_return	Prediction	% Diff Error
2	0	01/03/2015	0.017267645	0.0198265	115%
3	1	06/03/2015	0.002940277	0.002218154	75%
4	2	15/11/2015	-0.030255463	-0.027660225	91%
5	3	16/02/2016	0.004726262	0.012457117	264%
6	4	20/09/2016	0.007410575	0.007581918	102%
7	5	27/01/2018	0.016444916	0.022355165	136%
8	6	30/08/2018	-0.016996251	-0.013463508	79%
9	7	21/10/2018	0.005119918	0.004413606	86%
10	8	19/05/2019	0.069828161	0.083065202	119%
11	9	17/04/2020	0.020487468	0.019450719	95%
12					
13					

Bild: 10 unabhängige Samples für Vorhersage, Random Forest Regressor.

RMSE = 0.0138, 0.0165

Teil 3: Ergebnisse Bewerten - **Was waren die Hebel in unserem Projekt?**



Domain Know-How
Problem verstehen

Daten Modelieren
Verstehen und Feature Engineering

Daten Qualität
Man kriegt was man bezahlt...

Model Auswahl
Strategie und Erfahrungswerte

Model Tuning
Sampling
Hyper Parameter

Teil 4: Lessons Learned

Teil 4: Lessons Learned - **David: Was macht ein Modell “gut”?**

Business Perspektive

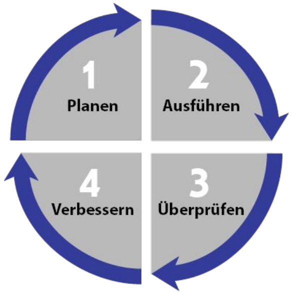


Bild: PDCA und kontinuierliche Verbesserung. Quelle

Entwickler Perspektive

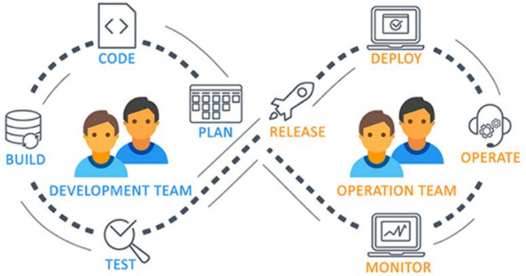


Bild: Development and operations. Quelle

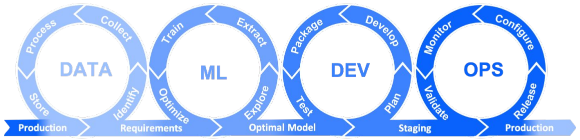


Bild: MLOPS, Integrating ML with DevOps. Quelle

IT Perspektive

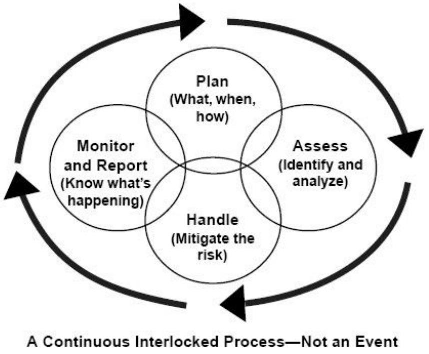


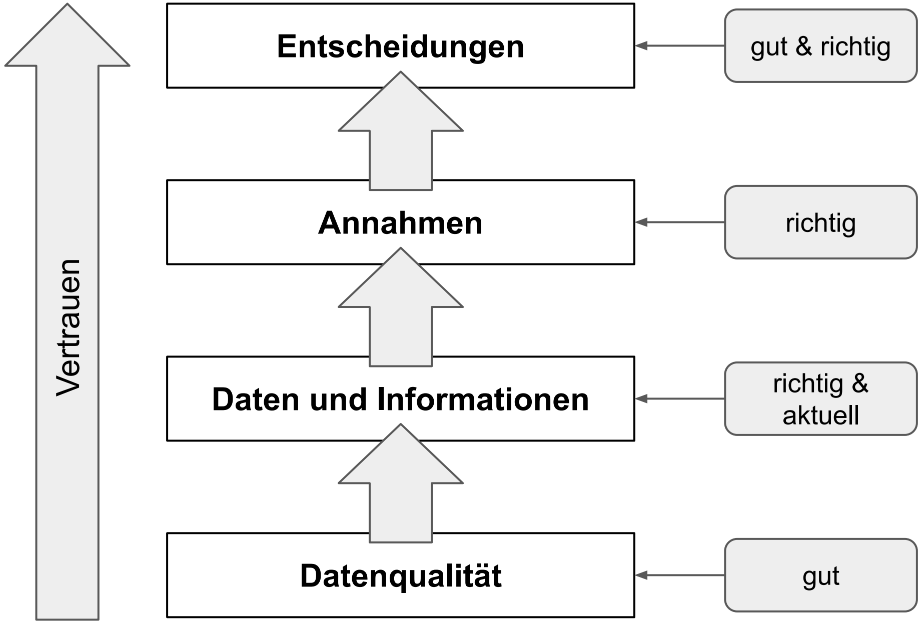
Bild: IT Risk Management. Quelle

<u>Frage an ZHAW:</u>
Was sind die typischen Risiken in einem ML Projekt?

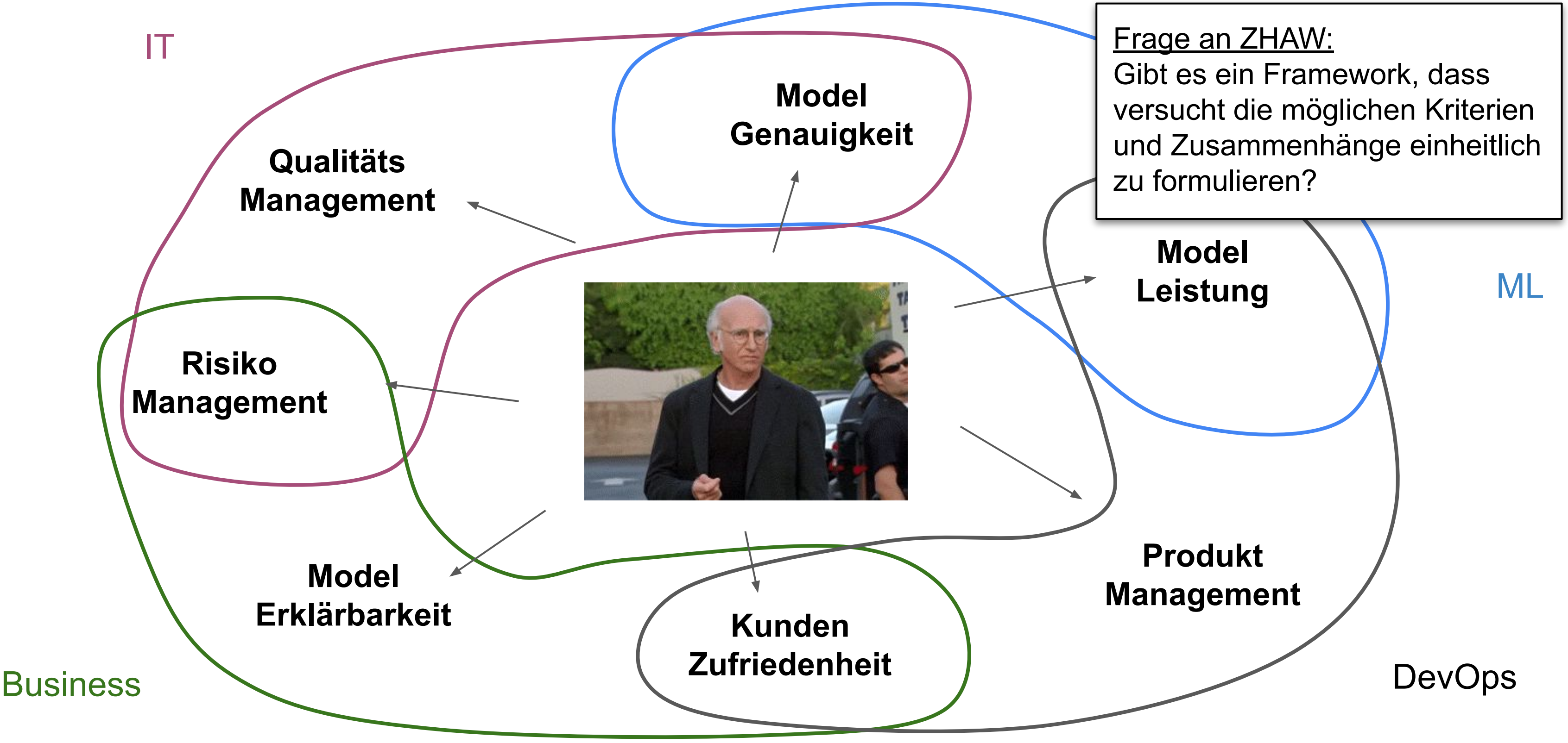
Teil 4: Lessons Learned - **David: Was macht ein Modell “gut”?**

Kunden Perspektive

<u>Frage an Alle:</u>
Ist **Vertrauen**
die neue **Währung** in
der IT Welt?



Teil 4: Lessons Learned - **David: Was macht ein Modell “gut”?**



Teil 4: Lessons Learned - **Akos**

Data is the new oil...

… but not all oil is refinable

Data quality is the key!

Teil 4: Lessons Learned - **Dave**

Danke für Eure Aufmerksamkeit!