Fallstudien II

Laura Kampmann, Christian Peters, Alina Stammen

12. Dezember 2020

Inhalt

Einleitung

Einleitung

Hier stehen ein paar Dinge über die Einleitung:

- Dies
- und
- das

Task I: Data Rate Prediction

Task I: Data Rate Prediction

hallo

Task I: Data Rate Prediction

Gradient Boosted Trees

Gradient Boosted Trees

- Kann man aus vielen ”schwachen” Lernern einen starken Lerner konstruieren?
  - $\Rightarrow$ Ja, Boosting ist eines der mächtigsten Konzepte des Machine Learning [2]
- Kombination von einfachen CART Bäumen zu einem starken Ensemble
  - $\Rightarrow$ Ähnlich zu Random Forest
- Der Unterschied zum Random Forest liegt im Training!

Training von Gradient Boosted Trees

- Bäume werden nacheinander zum Ensemble hinzugefügt
- Jeder neue Baum versucht, die Schwächen seiner Vorgänger ”auszubügeln”

  $\Rightarrow$ *Additives Training*
- Je mehr Bäume aufgenommen werden, desto geringer wird der Training-Error (das Modell wird aber komplexer)

  $\Rightarrow$ Kontrolle des *Bias-Variance Tradeoffs*

  $\Rightarrow$ Zusätzlich gibt es Regularisierungs-Parameter

Implementierung: XGBoost

- Liefert state-of-the-art Performance in einer Vielzahl von ML-Problemen
- In 2015 haben 19/25 Gewinner von Kaggle-Competitions XGBoost eingesetzt
- Kann problemlos auf mehrere Milliarden Training Samples skaliert werden
- Lässt sich aber auch hervorragend auf ressourcenbegrenzten Systemen einsetzen [1]

Task I: Data Rate Prediction

ARIMA

Aufgabenstellung Task I: Vorhersage der Datenrate

- weitere Ansätze zur Vorhersage der Zielgröße ”throughput”
- Aussagekraft der Einflussvariablen

Lösungsansätze

- XGboost
- ARMA Modell mit Regressionsfehlern

ARMA Modell mit Regressionsfehlern

Lineares Modell: $y = \beta \cdot X + \epsilon$, wobei ϵ Störfaktor und ϵ_i i.i.d.

- Problem: Autokorrelation
- Lösung: Anwendung des ARMA Modells auf die Regressionsfehler

ARMA Modell mit Regressionsfehlern

$ARMA(p, q)$: zusammengesetzes Modell aus

- $AR(p)$ (Auto Regressive): basiert auf vergangenen Werten ϵ_i des Response
- $MA(q)$ (Moving Average): basiert auf Fehlern e_i zwischen vergangenen Vorhersagen und wahrem Wert des Response
- Modellgleichung des ARMA Modells:

$$\epsilon_i = \underbrace{\phi_1\epsilon_{i-1} + ... + \phi_p\epsilon_{i-p}}_{AR(p)} \underbrace{-\theta_1 e_{i-1} - ... - \theta_q e_{i-q}}_{MA(q)} + \eta_i,$$

  mit η_i als Störfaktor

Wahl der Parameter p, q

- ACF (Autocorrelationfunktion) und PACF (partial Autocorrelationfunction) beschreiben die Korrelation der Lags mit dem aktuellen Zeitpunkt
- PACF beinhaltet nur direkte Einflüsse
- ACF dagegen betrachtet auch solche Einflüsse die indirekt sind
- die Funktionen legen damit die Wahl der Parameter p und q der Modell fest

hier könnte ein Bild sein

Modellgleichung ARMA mit Regressionsfehlern

Insgesamt ist die Modellgleichung gegeben durch

$$y_i = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p + \phi_1 \epsilon_{i-1} + ... + \phi_p \epsilon_{i-p}$$
$$-\theta_1 e_{i-1} - ... - \theta_q e_{i-q} + \eta_i$$

Task I: Data Rate Prediction

Validierung

frame

hallo

TaskII

frame

hallo

TaskII

DatentransformationTaskII

frame

hallo

TaskII

XGboostTaskII

frame

hallo

Irgendwas zum Schluss

Literatur i

T. Chen and C. Guestrin.
Xgboost: A scalable tree boosting system.
CoRR, abs/1603.02754, 2016.

T. Hastie, R. Tibshirani, and J. Friedman.
The elements of statistical learning: data mining, inference and prediction.
Springer, 2 edition, 2009.