

# Treffen mit Sebastian am Dienstag, 7.9. an der Uni Augsburg

## Ziel

- Abstecken der Ziele / Gliederung für die Masterarbeit
  - ➔ kann mit Schreiben beginnen
- Absprechen der weiteren Arbeit (Deliverables, wie zB Code)  
Zitationsstil: as stated in Sutton and Barto (2018). Bibliographie wie im Dokument  
„We“ als Erzähler
- Klären von Code-Feinheiten

## Nächste Schritte

- Recherche / Konzepte
  - o Wie Neuronales Netz einbauen
- Implementierung
  - o exponential smoothing überprüfen (theta\_k beeinflusst das sample v für die Berechnung von theta\_k^update)
  - o Neuronales Netz über Value Function Approximation einbauen
- Aufschreiben
  - o Weiteres Szenario im Netzwerkkontext
  - o CDLP
- Code
  - o API im Netzwerk
- Offene Fragen
  - o Neuronale Netze wie Architektur modellieren, wie Zsgh beschreiben

## Dokumente

- 0709 - LaTeX.pdf

## Erreicht

- Für single leg durchgerechnet (t=400, c=120), API dauert lange (15 h)
- ...

## Erkenntnisse

- CDLP Optimierung abhängig von Entscheidungsvariablen (leeres Offerset einschließen führt zu gleichem Zielfunktionswert bei unterschiedlicher Belegung der Entscheidungsvariablen)

## Ideen

- Kapazitätsbedarf von einem Produkt für eine Ressource in {0, 1, ..., N} statt nur in {0, 1}
- Preise sind bisher fixiert. Könnte man hier näher an die Realität (auch Preis für First Class ändert sich über Zeit)? Wird teilweise schon abgedeckt mit den versch. Produkten.

- Zeitwert des Geldes einführen. Der Fluggesellschaft sollte lieber sein, den Euro heute zu bekommen als morgen (könnte ihn investieren)  
=> Wertfunktion anpassen

## Große Punkte

- Rote Kästen im 0618 - 0 ADP
- API linear exponential smoothing  
Dokument: 0618 - 1 Exploring exponential smoothing (hier ohne epsilon-greedy)  
Dokument 0618 - 2 E\_API\_lin\_single\_leg with epsilon greedy  
exponential smoothing wie in Koch, 3.3. Update of parameters erwähnt, ist im Wesentlichen eine Durchschnittsbildung
- T-Test für Vergleich verschiedener Policies  
Code: T\_compare\_strategies\_single\_leg  
matched pairs t-test mit Null-Hypothese  $API - DP < 0$  (erwarte, dass DP besser ist)
- MLP Ansatz und wie an die Pi's kommen

## Kleine Punkte

- Sprache: A customer arrives. "He" oder "She" oder "It" buys a product.
- Arrival probability wird bei Bront et al über einen Poisson Prozess modelliert, was zu mehr als einem Customer pro Zeitschritt führen könnte. Im Erwartungswert  $\lambda$  pro Zeitschritt. Ich habe implementiert mit maximal 1 customer pro Zeitschritt (Summe der Ankunftsweiten  $< 1$ ), relevant bei `simulate_sales(offer_set)`.
- LaTeX-Vorlage am Lehrstuhl für zB richtigen Zeilenabstand?

## Neue Punkte

- API: warum dieselben sample paths in jeder policy iteration?

## Nächstes Treffen

- Notation (zB t Index oben rechts)
-