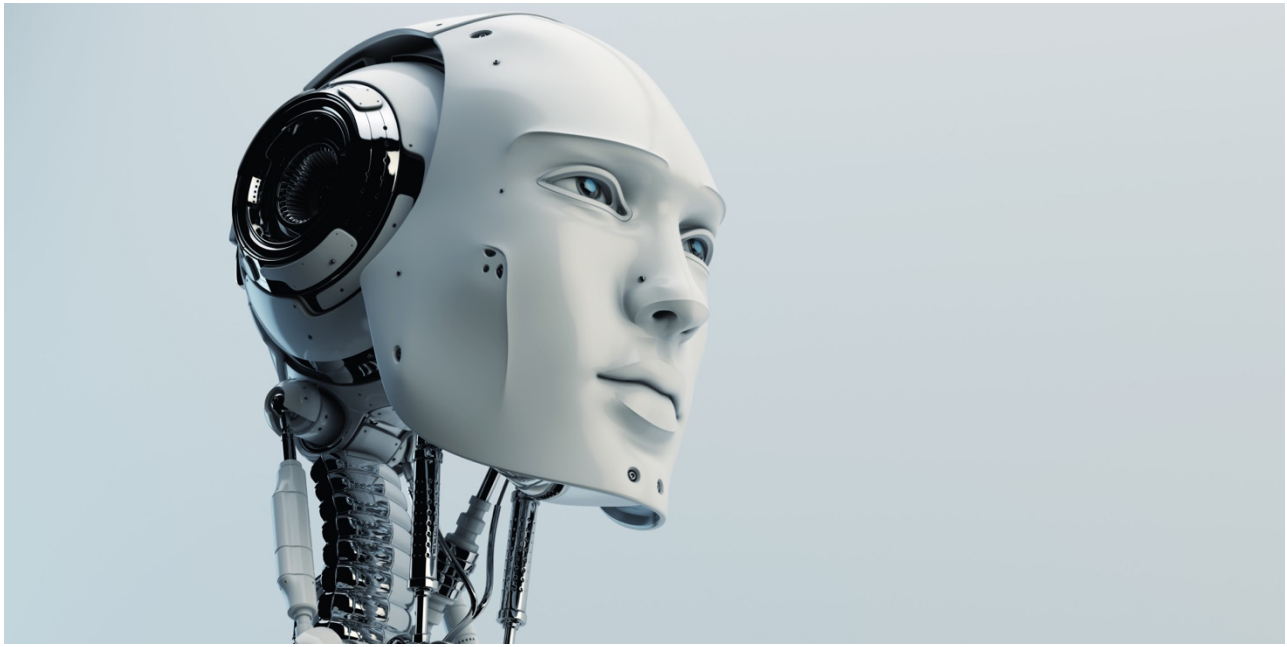
(Online) Least-Squares Policy Iteration

TECHNISCHE UNIVERSITÄT DARMSTADT

Seminar aus Künstlicher Intelligenz

Albert Slawik



Bildquelle: http://i.huffpost.com/gen/1803076/images/o-AI-facebook.jpg

Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Einleitung**
- Relevante Arbeiten
- Methode der kleinsten Quadrate
- Least Squares Temporal Difference
- Incremental Least Squares Temporal Difference
- Ergebnisse
- Diskussion

Einleitung

TECHNISCHE UNIVERSITÄT DARMSTADT

Wird zur Lösung von Reinforcement Learning Problemen genutzt

Bei großen Problemen lassen sich policies nur approximieren

Temporal Difference zwar schnell aber ungenau und “vergesslich”

LSTD löst Value Function explizit

Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- **Relevante Arbeiten**
- Methode der kleinsten Quadrate
- Least Squares Temporal Difference
- Incremental Least Squares Temporal Difference
- Ergebnisse
- Diskussion

Relevante Arbeiten

TECHNISCHE UNIVERSITÄT DARMSTADT

- S. Bradtke, A. Barto – Linear Least-Squares Algorithms for Temporal Difference Learning
- J. A. Boyan – Least-Squares Temporal Difference Learning
- R. S. Sutton et al. – Incremental Least-Squares Temporal Difference Learning
- R. S. Sutton et al. – iLSTD: Eligibility Traces and Convergence Analysis

Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- Relevante Arbeiten
- **Methode der kleinsten Quadrate**
- Least Squares Temporal Difference
- Incremental Least Squares Temporal Difference
- Ergebnisse
- Diskussion

Methode der kleinsten Quadrate

TECHNISCHE UNIVERSITÄT DARMSTADT

Messwerte: y_i

Modellkurve: $f(x_i)$

Modellparameter:

$$\vec{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{R}^m$$

Modellkurvenvektor:

$$\vec{f} = (f(x_1, \vec{\alpha}), \ldots, f(x_n, \vec{\alpha})) \in \mathbb{R}^n$$

Methode der kleinsten Quadrate

TECHNISCHE UNIVERSITÄT DARMSTADT

Fehlerquadratsumme:

$$\sum_{i=1}^{n}(f(x_i,\vec{\alpha})-y_i)^2 = \|\vec{f}-\vec{y}\|_2^2.$$

Wähle Parameter, sodass:

$$\min_{\vec{\alpha}} \|\vec{f}-\vec{y}\|_2^2.$$

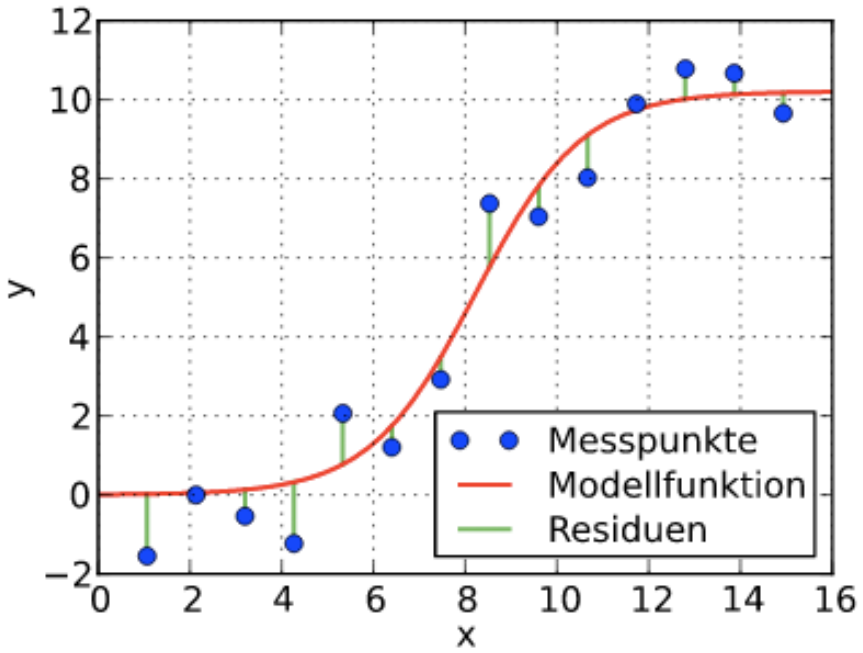


Abbildung 1: Messpunkte mit durch Least Squares bestimmter Modellfunktion

Bildquelle: https://de.wikipedia.org/wiki/Methode_der_kleinsten_Quadrate

Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- Relevante Arbeiten
- Methode der kleinsten Quadrate
- **Least Squares Temporal Difference**
- Incremental Least Squares Temporal Difference
- Ergebnisse
- Diskussion

Least-Squares Temporal Difference

TECHNISCHE UNIVERSITÄT DARMSTADT

Ansatz gleich zu TD

Approximation der Value Function:

$$V_{\boldsymbol{\theta}}(s) = \boldsymbol{\phi}(s)^T \boldsymbol{\theta}$$

Feature Representation des Zustandraumes:

$$\boldsymbol{\phi} : \mathcal{S} \rightarrow \Re^n$$

Parameter der Value Function:

$$\boldsymbol{\theta} \in \Re^n$$

Least-Squares Temporal Difference

TECHNISCHE UNIVERSITÄT DARMSTADT

Parameter Update nach TD:

$$\begin{aligned} \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t + \alpha_t \mathbf{u}_t(\boldsymbol{\theta}_t), \text{ where} \\ \mathbf{u}_t(\boldsymbol{\theta}) &= \boldsymbol{\phi}(s_t)\delta_t(V_{\boldsymbol{\theta}}). \end{aligned}$$

Dabei ist α_t die Learning Rate

$V_{\boldsymbol{\theta}}$ der geschätzte Wert in Abhängigkeit von θ

und $\mathbf{u}_t(\boldsymbol{\theta})$ das TD Update zum Zeitpunkt *t*

$\delta_t(V)$ ist der TD Fehler und Definiert als:

$$\delta_t(V) = r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$$

Least-Squares Temporal Difference

TECHNISCHE UNIVERSITÄT DARMSTADT

LSTD löst explizit nach den Parametern der Value Function

Dazu wird $\mu_t(\theta)$ als Summe der TD Updates definiert

$$\boldsymbol{\mu}_t(\boldsymbol{\theta}) \quad = \quad \sum_{i=1}^{t} \mathbf{u}_i(\boldsymbol{\theta})$$

Da das Update einem Gradienten entspricht, reicht es, die Summe für das Minimum Null zu setzen

Least-Squares Temporal Difference

TECHNISCHE UNIVERSITÄT DARMSTADT

Umformung ergibt:

$$
\begin{aligned}
\boldsymbol{\mu}_t(\boldsymbol{\theta}) &= \sum_{i=1}^{t} \boldsymbol{\phi}_t \delta_t(V_{\boldsymbol{\theta}}) \\
&= \sum_{i=1}^{t} \boldsymbol{\phi}_t \left(r_{t+1} + \gamma \boldsymbol{\phi}_{t+1}^T \boldsymbol{\theta} - \boldsymbol{\phi}_t^T \boldsymbol{\theta} \right) \\
&= \sum_{i=1}^{t} \left(\boldsymbol{\phi}_t r_{t+1} - \boldsymbol{\phi}_t (\boldsymbol{\phi}_t - \gamma \boldsymbol{\phi}_{t+1})^T \boldsymbol{\theta} \right) \\
&= \Bigg(\underbrace{\sum_{i=1}^{t} \boldsymbol{\phi}_t r_{t+1}}_{\mathbf{b}_t} - \underbrace{\sum_{i=1}^{t} \boldsymbol{\phi}_t (\boldsymbol{\phi}_t - \gamma \boldsymbol{\phi}_{t+1})^T}_{\mathbf{A}_t} \boldsymbol{\theta} \Bigg) \\
&= (\mathbf{b}_t - \mathbf{A}_t \boldsymbol{\theta}).
\end{aligned}
$$

Da $\mu_t(\theta) = 0$, folgt:

$$\boldsymbol{\theta}_{t+1} = \mathbf{A}_t^{-1} \mathbf{b}_t$$

Least-Squares Temporal Difference

TECHNISCHE UNIVERSITÄT DARMSTADT

0 $s \leftarrow s_0$, $\mathbf{A} \leftarrow \mathbf{0}$, $\mathbf{b} \leftarrow \mathbf{0}$
1 Initialize $\boldsymbol{\theta}$ arbitrarily
2 **repeat**
3 Take action according to π and observe r, s'
5 $\mathbf{b} \leftarrow \mathbf{b} + \boldsymbol{\phi}(s)r$
6 $\mathbf{d} \leftarrow (\boldsymbol{\phi}(s) - \gamma\boldsymbol{\phi}(s'))$
7 $\mathbf{A} \leftarrow \mathbf{A} + \boldsymbol{\phi}(s)\mathbf{d}^T$
8 **if** (first update)
9 $\tilde{\mathbf{A}} \leftarrow \mathbf{A}^{-1}$
10 **else**
11 $\tilde{\mathbf{A}} \leftarrow \tilde{\mathbf{A}}\left(I - \left(\frac{\boldsymbol{\phi}(s)\mathbf{d}^T}{1 + \mathbf{d}^T\tilde{\mathbf{A}}\boldsymbol{\phi}(s)}\right)\tilde{\mathbf{A}}\right)$
12 **end if**
13 $\boldsymbol{\theta} \leftarrow \tilde{\mathbf{A}}\mathbf{b}$
14 **end repeat**

Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- Relevante Arbeiten
- Methode der kleinsten Quadrate
- Least Squares Temporal Difference
- **Incremental Least Squares Temporal Difference**
- Ergebnisse
- Diskussion

iLSTD

TECHNISCHE UNIVERSITÄT DARMSTADT

Ziel: inkrementelles Berechnen von $\mu_t(\theta)$ nachdem eine Transition beobachtet wurde und θ sich geändert hat

Löst nicht direkt nach den Parametern, die Null ergeben

Vergleichbar mit Gradienten Verfahren, die sich in Richtung Minimum bewegen

Dazu werden b_t und A_t inkrementell in Abhängigkeit des beobachteten Rewards berechnet:

$$\mathbf{b}_t = \mathbf{b}_{t-1} + \underbrace{r_t \boldsymbol{\phi}_t}_{\Delta \mathbf{b}_t}$$

$$\mathbf{A}_t = \mathbf{A}_{t-1} + \underbrace{\boldsymbol{\phi}_t (\boldsymbol{\phi}_t - \gamma \boldsymbol{\phi}_{t+1})^T}_{\Delta \mathbf{A}_t}$$

iLSTD

TECHNISCHE UNIVERSITÄT DARMSTADT

Nun kann man die Summe der TD Updates des aktuellen Zeitschrittes inkrementell berechnen:

$$\boldsymbol{\mu}_t(\boldsymbol{\theta}_t) \quad = \quad \boldsymbol{\mu}_{t-1}(\boldsymbol{\theta}_t) + \Delta \mathbf{b}_t - (\Delta \mathbf{A}_t)\boldsymbol{\theta}_t$$

Wenn nun gilt:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_t$$

Kann man das TD Updates des nächsten Zeitschritts ebenfalls inkrementell berechnen:

$$\boldsymbol{\mu}_t(\boldsymbol{\theta}_{t+1}) \quad = \quad \boldsymbol{\mu}_t(\boldsymbol{\theta}_t) - \mathbf{A}_t(\Delta\boldsymbol{\theta}_t)$$

iLSTD

TECHNISCHE UNIVERSITÄT DARMSTADT

Direktes lösen von $\mu_t(\theta_{t+1}) = 0$ immer noch sehr aufwändig

Berechnung von $\Delta\theta_t$ ebefalls nur in $O(n^2)$ machbar

Lösung: Ergebnisse aus dem letzten Zeitschritt behalten und nur *m* einzelne Komponenten neu berechnen:

$$\begin{aligned} \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t + \alpha_t \mu_t(i) \boldsymbol{e}_i \\ \boldsymbol{\mu}_t(\boldsymbol{\theta}_{t+1}) &= \boldsymbol{\mu}_t(\boldsymbol{\theta}_t) - \alpha_t \mu_t(i) \mathbf{A}_t \boldsymbol{e}_i \end{aligned}$$

iLSTD

TECHNISCHE UNIVERSITÄT DARMSTADT

0 $s \leftarrow s_0$, $\mathbf{A} \leftarrow \mathbf{0}$, $\boldsymbol{\mu} \leftarrow \mathbf{0}$, $t \leftarrow 0$
1 Initialize $\boldsymbol{\theta}$ arbitrarily
2 **repeat**
3 Take action according to π and observe r, s'
4 $t \leftarrow t + 1$
5 $\Delta\mathbf{b} \leftarrow \boldsymbol{\phi}(s)r$
6 $\Delta\mathbf{A} \leftarrow \boldsymbol{\phi}(s)(\boldsymbol{\phi}(s) - \gamma\boldsymbol{\phi}(s'))^T$
7 $\mathbf{A} \leftarrow \mathbf{A} + \Delta\mathbf{A}$
8 $\boldsymbol{\mu} \leftarrow \boldsymbol{\mu} + \Delta\mathbf{b} - (\Delta\mathbf{A})\boldsymbol{\theta}$
9 **for** i from 1 to m **do**
10 $j \leftarrow \text{argmax}(|\mu_j|)$
11 $\theta_j \leftarrow \theta_j + \alpha\mu_j$
12 $\boldsymbol{\mu} \leftarrow \boldsymbol{\mu} - \alpha\mu_j\mathbf{A}\boldsymbol{e}_i$
13 **end for**
14 **end repeat**

Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- Relevante Arbeiten
- Methode der kleinsten Quadrate
- Least Squares Temporal Difference
- Incremental Least Squares Temporal Difference
- **Ergebnisse**
- Diskussion

Ergebnisse

TECHNISCHE UNIVERSITÄT DARMSTADT

TD: $O(n)$, wenn dünnbesetzt $O(k)$
LSTD: $O(n^2)$ wenn Matrix inkrementell invertiert wird, sonst $O(n^3)$
iLSTD: $O(mn + k^2)$

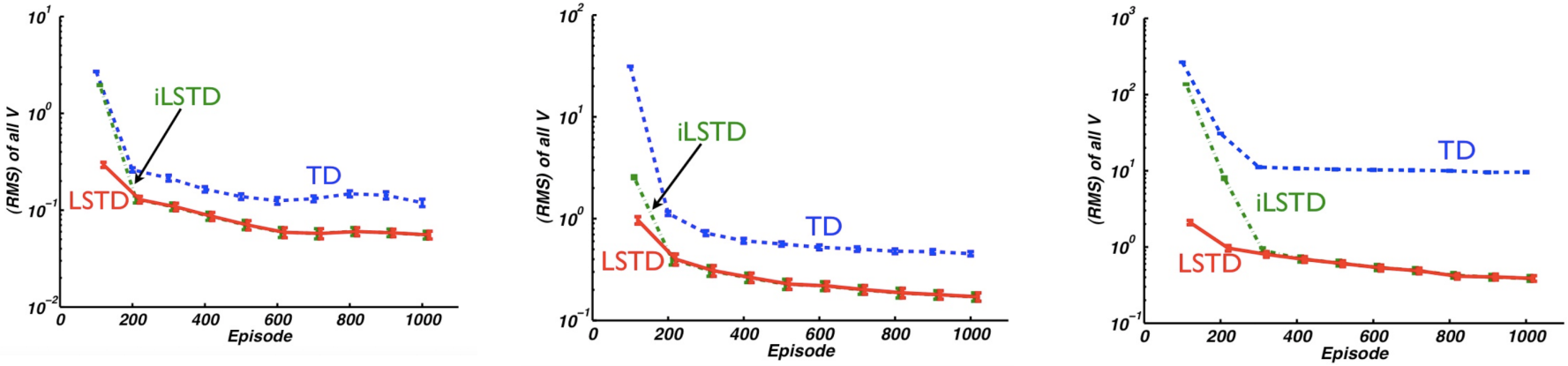
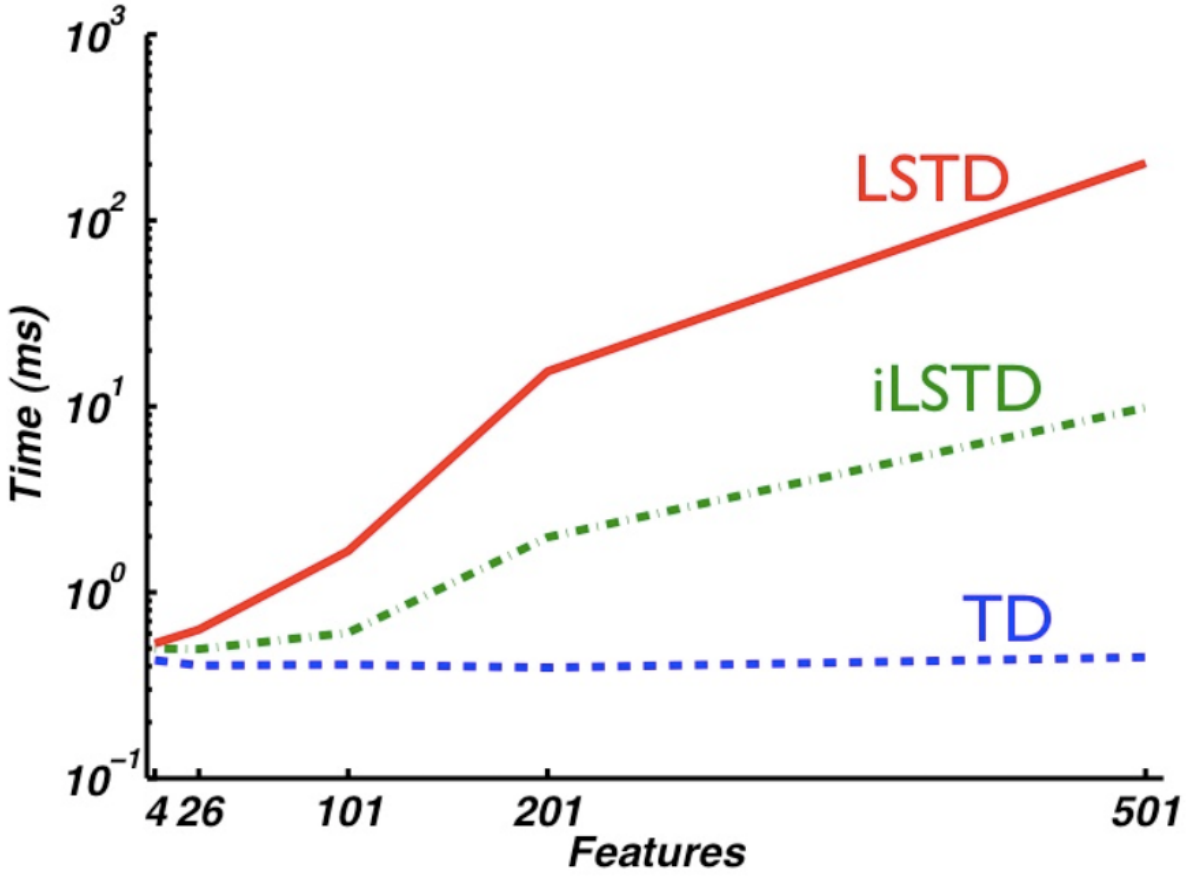


Abbildung 2: Fehlerverhalten bei unterschiedlichen Problemgrößen

Ergebnisse

TECHNISCHE UNIVERSITÄT DARMSTADT



Inhalt

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- Relevante Arbeiten
- Methode der kleinsten Quadrate
- Least Squares Temporal Difference
- Incremental Least Squares Temporal Difference
- Ergebnisse
- **Diskussion**

Diskussion

TECHNISCHE UNIVERSITÄT DARMSTADT

Erweiterung von iLSTD auf iLSTD(λ)

Konvergenzbeweis von iLSTD

Anwendung in komplexen Applikationen wie Soccer Keepaway

TECHNISCHE UNIVERSITÄT DARMSTADT

Vielen Dank für Ihre Aufmerksamkeit!

Appendix

TECHNISCHE UNIVERSITÄT DARMSTADT

Boyan Chain: