

Vortrag im Rahmen des Seminars:

„Artificial Life und Multiagentensysteme“

Prof. Dr. Winfried Kurth
Sommersemester 2003

Prognose von Zeitreihen mit GA/GP

Mathias Radicke, Informatikstudent, 10. Semester

Gliederung

1. Einleitung
2. KI-Methoden zur Zeitreihenprognose
 - 2.1. Genetische Programmierung für Handelsentscheidungen
 - 2.2. Genetische Algorithmen für Handelsentscheidungen
 - 2.3. Neuronale Netze für Kursvorhersagen
 - 2.4. Genetische Programmierung für ökonomische Formeln
3. Vorhersage von Fußballergebnissen mit GP



1. Einleitung



Was will man mit der Prognose von Zeitreihen erreichen?

- Aktien-, Devisen-, Indexkursentwicklung vorhersagen
- Handelsentscheidungen für Wertpapiere treffen
- ökonomische Entwicklungen vorhersagen (Geldmenge, Inflation, ...)
- (Vorhersage von Fußballergebnissen)

Warum GA/GP zur Prognose von Zeitreihen?

- unzählige Eingabedaten, großer Suchraum
- unsystematische Eingabeparameter
- fehlende Lösungsansätze, -strategien

2. KI-Methoden zur Zeitreihenprognose



Welche Methoden nutzt man zur Prognose von Zeitreihen?

- Genetische Programmierung für Handelsentscheidungen
- Genetische Algorithmen für Handelsentscheidungen, Optimierung
- Neuronale Netze für längerfristige Kursvorhersagen (optimale Parameterbelegung des Neuronalen Netzes durch GA)
- Genetische Programmierung für „einfache“ ökonomische Formeln (für empirische Daten)

2.1. Genetische Programmierung für Handelsentscheidungen



„GA/GP Trading Rule Optimizer“¹

- Echtzeit-Handelssystem für US Dollar/Britisches Pfund
- 1. Problemrepräsentation für GP:

IF (Function 1 evaluated at Frequency 1) = TRUE

AND (Function 2 evaluated at Frequency 2) = TRUE

THEN *Buy US T-Bond Futures*

- 2. Fitness-Funktion:
Profitabilität der gefundenen Regel für historische Kursdaten
- 3. Start-Population:
Zufällig ausgewählt

¹ www-cfr.jims.cam.ac.uk/archive/PAPERS/2000/geneticprogramming.pdf

2.1. Genetische Programmierung für Handelsentscheidungen



„GA/GP Trading Rule Optimizer“

GP/GA Trading Rule Optimizer

GA/GP Trading Rule Optimizer

UNIVERSITY OF CAMBRIDGE
The Judge Institute of Management Studies

System Parameters

Indicators to Use

- AMA
- DCI
- MACD
- MA Crossover
- Momentum Oscillator
- Price Channel
- Reversals
- RSI
- Stochastics

Indicator Frequency

- Daily
- 8-Hour
- 4-Hour
- Hour
- Q.Hour
- Minute

Indicator Horizon

	Max.Lag	Max.Lag	Max.Lag
<input type="checkbox"/> Daily	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="checkbox"/> 8-Hour	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="checkbox"/> 4-Hour	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="checkbox"/> Hour	<input type="text" value="0"/>	<input checked="" type="checkbox"/> Q.Hour	<input type="text" value="0"/>
<input type="checkbox"/> Minute	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

Trading Parameters

- Fixed Slippage
- Variable Slippage
- Always in Market
- Drewdown Limit

Time Horizon

Mining

Year	Month	Day	Hour	Minute
From 1994	01	01	00	00
To 1994	03	31	23	59

Testing

Year	Month	Day	Hour	Minute
From 1994	04	01	00	00
To 1994	06	30	23	59

Log File Location

- Out of Sample Only

GA/GP Parameters

Algorithm

- Genetic Programming
- Genetic Algorithm

Maximum Indicators

Minimum Iterations Maximum Iterations

Crossover Rate Mutation Rate

Population

Boolean Operators

- AND
- EQV
- OR
- IMP
- XOR

OK Cancel

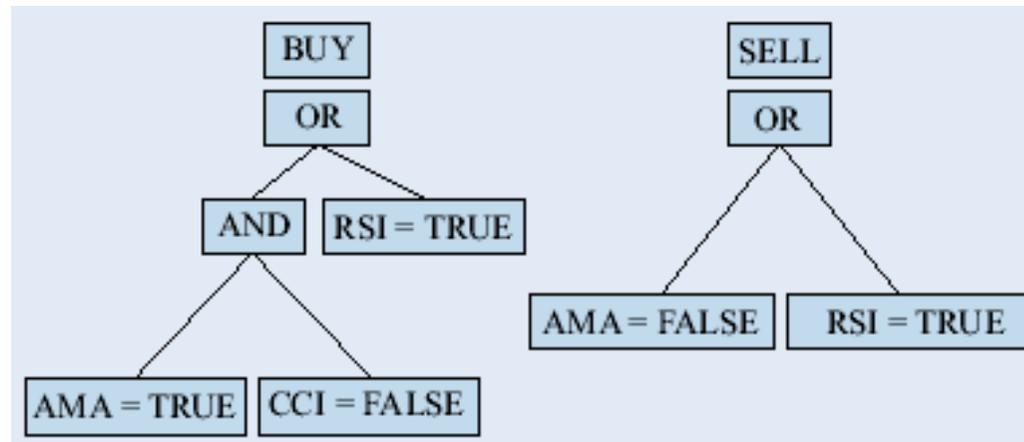
2.1. Genetische Programmierung für Handelsentscheidungen



„GA/GP Trading Rule Optimizer“

Funktionsweise:

- Terminalsymbole sind technische Indikatoren
- Funktionen sind boolesche Operatoren
- Selektion findet statt nach Profitabilität und Verlustrisiko
- Test findet auf „Out-Of-Sample“-Daten oder Echtzeitdaten statt
- Beispielergebnis:



2.1. Genetische Programmierung für Handelsentscheidungen



„GA/GP Trading Rule Optimizer“

Ergebnisse:

- Profitabilität steigt und Verlustrisiko sinkt von Generation zu Generation
- die besten 20 Individuen bilden eine Portfoliostrategie:
 - getestet an 15 „Out-Of-Sample“ Quartalen
 - Rendite von 4,8 % p.a.
 - das beste Individuum erzielte 7 % p.a.
- Strategie wird schlechter, je weiter die Testdaten von den Trainingsdaten entfernt sind
- → adaptives System, das sich bei Verschlechterung neu berechnet

2.2. Genetische Algorithmen für Handelsentscheidungen

Anwendungen:

- „optimale“ Depotzusammensetzung berechnen
- 1. Problemrepräsentation für GA:
Gene sind mehrere Tausend verschiedene Wertpapiere
- 2. Fitness-Funktion:
Profitabilität und Verlustrisiko des Depots
- 3. Start-Population:
zufällig ausgewählt

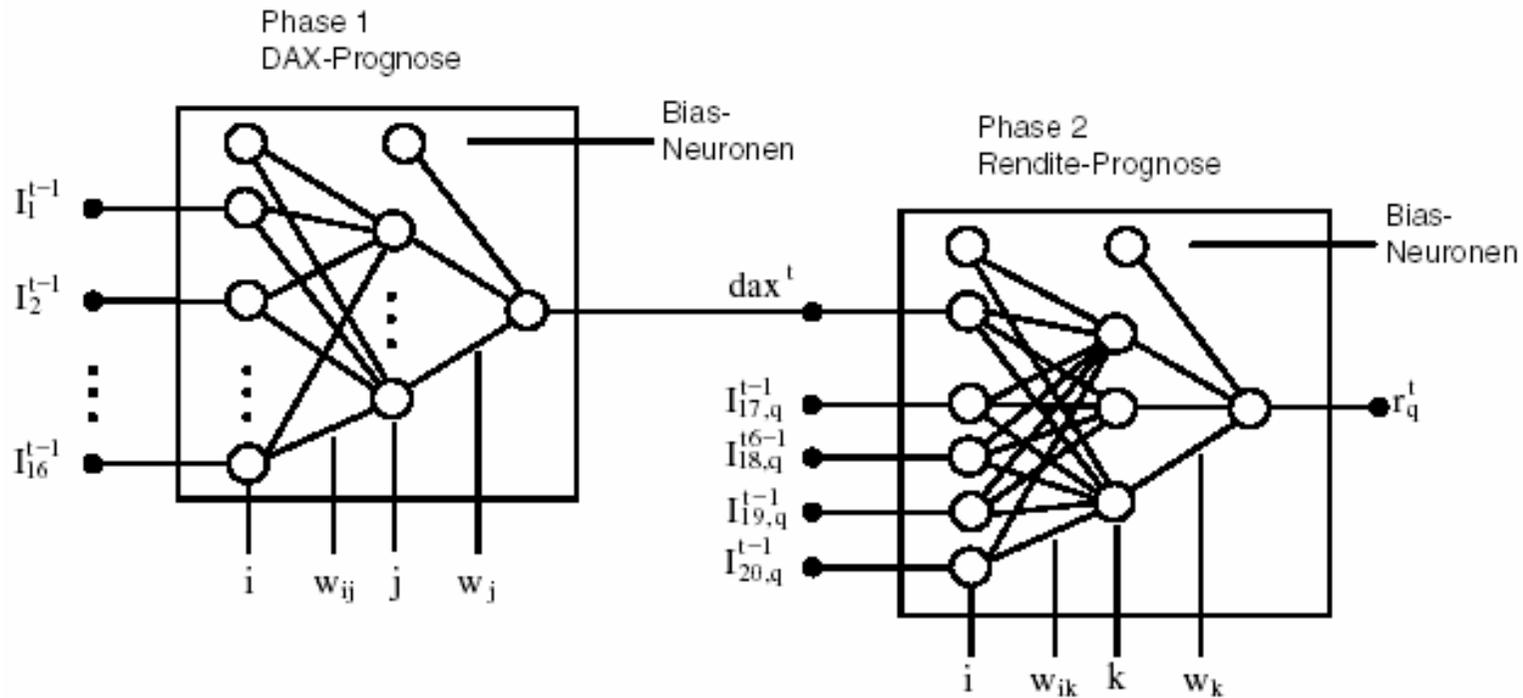


2.3. Neuronale Netze für Kursvorhersagen



Funktionsweise ²:

- neuronales Netz für die Kurs- und Risikovorhersage



² www.fernuni-hagen.de/FBWIWI/DP/DP268.pdf

2.3. Neuronale Netze für Kursvorhersagen



Funktionsweise:

- GA zur Optimierung der Topologie und/oder Parameterwerte des Vorhersagenetzwerkes (z.B. Lernrate, Anzahl versteckter Neuronen, ...)
- 1. Problemrepräsentation für GA:
 - Gene sind zusammengesetzt aus Aktivitätsschwellwerten der Bias-Neuronen, aus Verbindungsgewichten und aus Parametern der Portfolio-Optimierung
- 2. Fitness-Funktion:
 - Risikonutzen des Anlagesystems
- 3. Start-Population:
 - zufällig ausgewählt

2.4. Genetische Programmierung für ökonomische Formeln



Funktionsweise³:

- Beispiel: „Verkehrsgleichung“ $M * V = Y * P$
M: Geldmenge, V: Umlaufgeschwindigkeit, Y: BIP, P: Preisniveau
- 1. Problemrepräsentation für GP:
Terminalsymbole: Y, M, i (Zinsrate), r (Zufallszahl [-1.000, 1.000])
Funktionssymbole: +, -, *, %, EXP, RLOG
- 2. Fitness-Funktion:
Standardabweichung von empirischen Daten
- 3. Start-Population:
zufällig ausgewählt

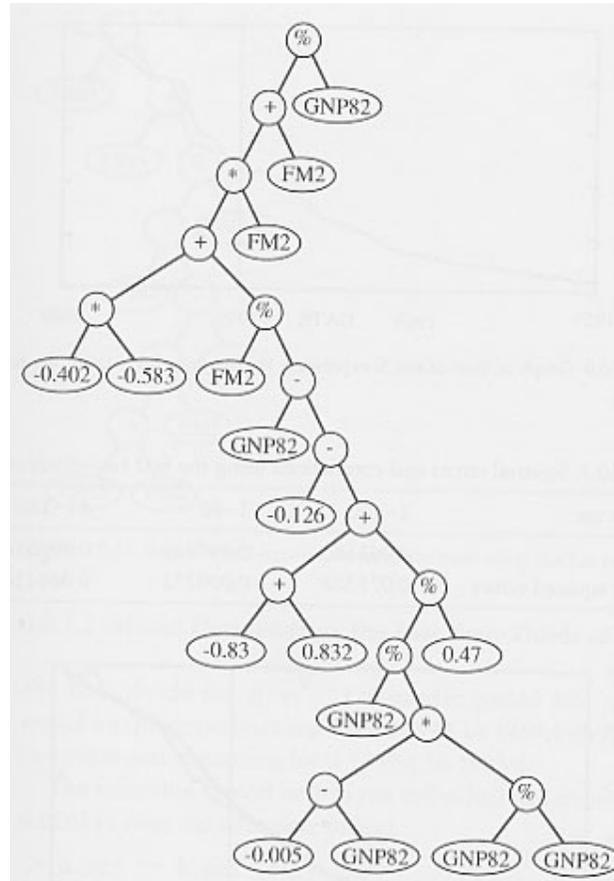
³ Genetic Programming, John R. Koza, Kapitel 10.3 “ECONOMETRIC MODELING AND FORECASTING”

2.4. Genetische Programmierung für ökonomische Formeln



Ergebnis:

- GNP82: Y
- FM2: M



- „Verkehrsgleichung“ $M * 1.634 = Y * P$

3. Vorhersage von Fußballergebnissen mit GP



Versuchsaufbau allgemein:

- Eingabedaten: alle Ergebnisse der 1. Fußball-Bundesliga seit 1996
- Testläufe:
 - alle Ergebnisse des 34. Spieltags von 1996 bis 2001 als Trainingsdaten
 - alle Ergebnisse des 34. Spieltags von 2002 als Testdaten („Out-Of-Sample“)
 - max. 54 Treffer in den Trainingsdaten
 - max. 9 Treffer in den Testdaten
 - Dauer der Trainingsläufe 50 Generationen
- Framework: „ECJ 9“ <http://www.cs.umd.edu/projects/plus/ec/ecj/>

3. Vorhersage von Fußballergebnissen mit GP



Programmaufbau (erster Versuch):

- 1. Problemrepräsentation für GP:

Terminalsymbole: „Heimstärke der Heimmannschaft“, „Auswärtsstärke der Heimmannschaft“, „Heimstärke der Gastmannschaft“, „Auswärtsstärke der Gastmannschaft“, letzte 3 „Heimergebnisse der Heimmannschaft“, letzte 3 „Auswärtsergebnisse der Heimmannschaft“, ..., 0, 1, -1

Funktionssymbole: +, -, *, /

- 2. Fitness-Funktion:

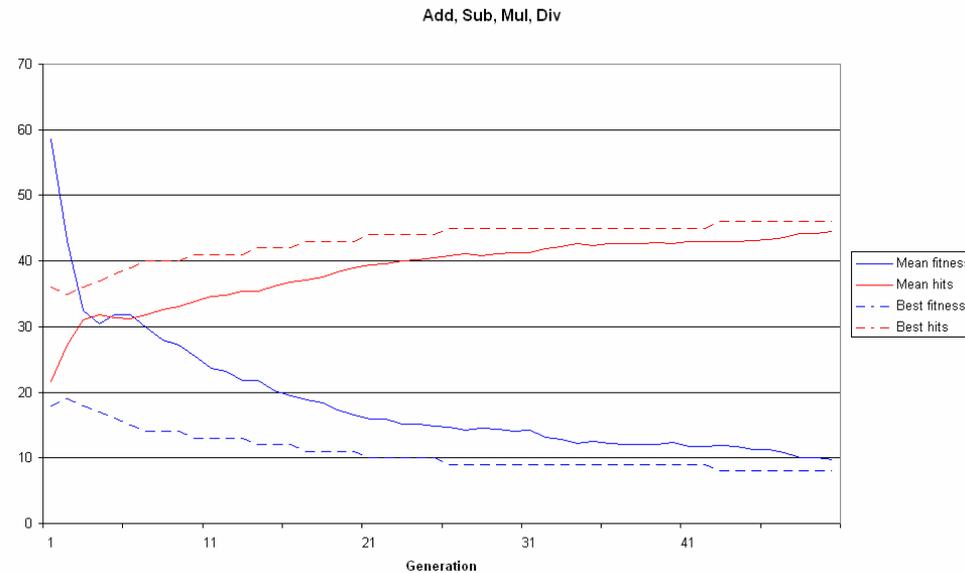
Anzahl der richtigen Vorhersagen ($x < 0$ = Niederlage der Heimmannschaft,
 $x > 0$ = Sieg der Heimmannschaft)

- 3. Start-Population: zufällig ausgewählt

3. Vorhersage von Fußballergebnissen mit GP



Ergebnisse (erster Versuch):



- Trainingsdaten - max. 46 Treffer
 - Testdaten - bestes Individuum erzielte 6 Treffer (Ausnahme!!!)
 - Ergebnisse für Unentschieden ungeeignet
- Umstellung des Programms auf boolesche Algebra

3. Vorhersage von Fußballergebnissen mit GP



Programmaufbau (zweiter Versuch):

- 1. Problemrepräsentation für GP:

Terminalsymbole: letzte 3 „Heimergebnisse der Heimmannschaft“,

letzte 3 „Auswärtsergebnisse der Heimmannschaft“, letzte 6 „Ergebnisse der Heimmannschaft“, letzte 6 „Ergebnisse der Auswärtsmannschaft“, ..., FALSE, TRUE

Funktionssymbole: NOT, OR, AND, XOR

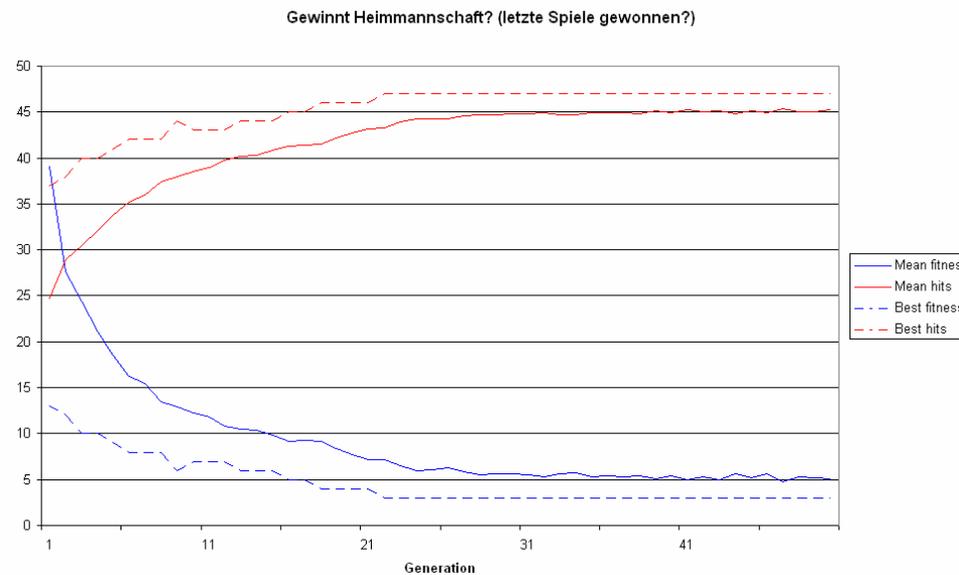
- 2. Fitness-Funktion:

Anzahl der richtigen Vorhersagen (FALSE = kein Sieg der Heimmannschaft,
TRUE = Sieg der Heimmannschaft)

- 3. Start-Population: zufällig ausgewählt

3. Vorhersage von Fußballergebnissen mit GP

Ergebnisse (zweiter Versuch):



- Fitness steigt schneller als beim ersten Versuch !!!
- Trainingsdaten - max. 47 Treffer
- Testdaten - bestes Individuum erzielte 5 Treffer

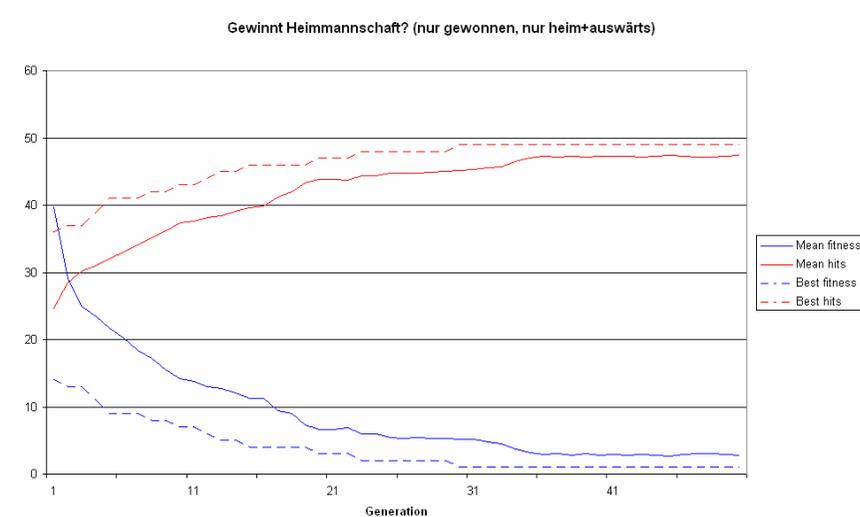
3. Vorhersage von Fußballergebnissen mit GP



Feintuning am zweiten Modell:

- nur letzte 3 Heim-/Auswärtsergebnisse der Heim-/Auswärtsmannschaft (letzte 6 Ergebnisse weggelassen)

→ noch schnellerer Fitnessanstieg



- letzte 4 Heim-/Auswärtsergebnisse der Heim-/Auswärtsmannschaft

→ nicht so schneller Fitnessanstieg

3. Vorhersage von Fußballergebnissen mit GP



Feintuning am zweiten Modell:

- Fitnessfunktion geändert:

von:

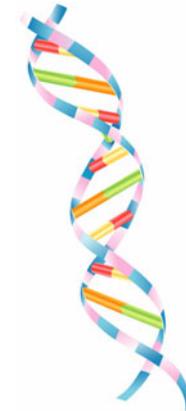
```
if ( mehr als 4 Treffer an einem Spieltag )  
then ( 9 - Treffer )  
else ( 9 - Treffer ) * 2          // Strafpunkte
```

in:

```
then ( 9 - Treffer ) * ( 9 - Treffer )  
          // quadratische Strafe
```

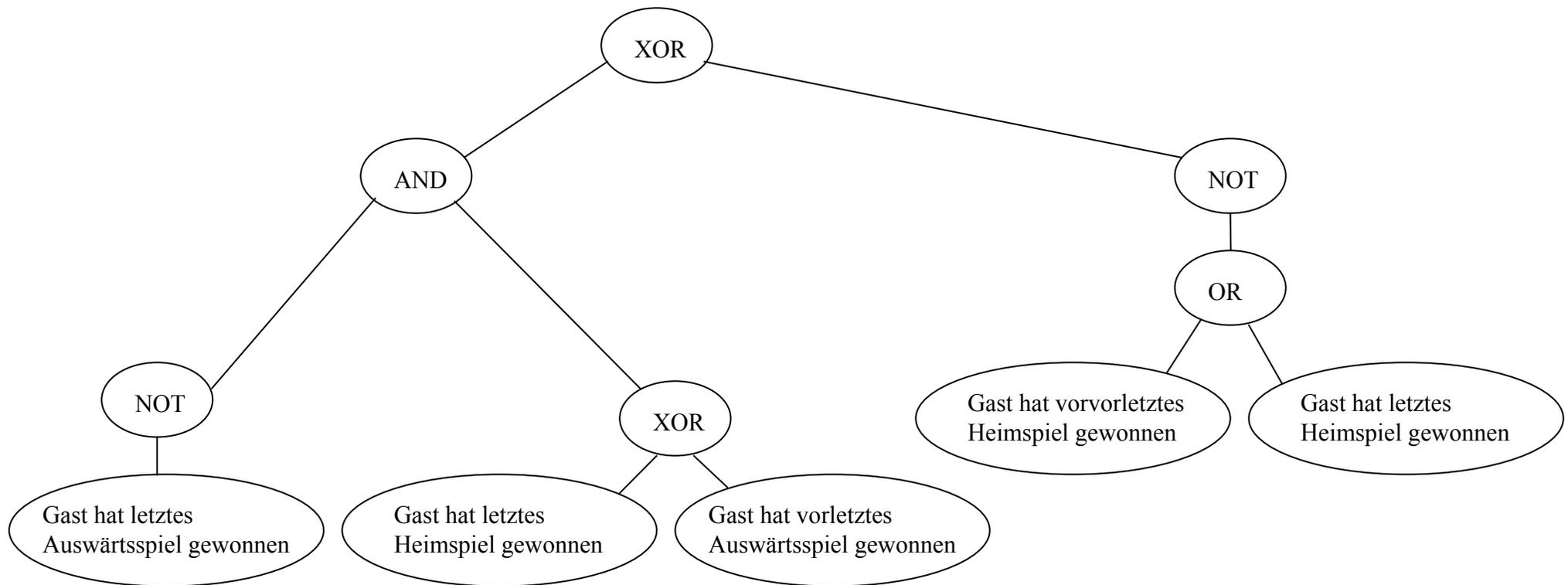
- noch schnellerer Fitnessanstieg
- im Training bis zu 53 Treffer
- im Test jedoch schlechter, 3 Treffer

3. Vorhersage von Fußballergebnissen mit GP



Ergebnis nach einer Generation (Beispiel):

- (XOR (AND (NOT gastgast1) (XOR gastheim1 gastgast2)) (NOT (OR gastheim3 gastheim1)))



3. Vorhersage von Fußballergebnissen mit GP

Fazit:

- je schneller die Fitness im Training steigt, desto schlechter scheinen die Ergebnisse im Test zu sein (Überzüchtung???)
- Ergebnisse insgesamt nicht befriedigend (durchschnittlich 4 - 5 Treffer)
- vielleicht zu wenig Trainingsdaten, schlechte Fitnessfunktion oder ungeeignete Verwendung von Terminal- oder Funktionssymbolen

