Computational Intelligence

Einkriterielle Evolutionäre Algorithmen II

Dr. Carsten Franke

Züricher Hochschule für Angewandte Wissenschaften

24. Februar 2015

Vorlesungsplanung - Termine (Änderungen vorbehalten)

1. 17.02.2015: Einkriterielle Evolutionäre Optimierung I (CF)
2. 24.02.2015: Einkriterielle Evolutionäre Optimierung II (CF)
3. 03.03.2015: Test (1+2), Mehrkriterielle Evolutionäre Optimierung I (CF)
4. 10.03.2015: Mehrkriterielle Evolutionäre Optimierung II (CF)
5. 17.03.2015: Statistische Lerntheorie I (JP)
6. 24.03.2015: Statistische Lerntheorie II (JP)
7. 21.04.2015: Test (5+6), Neuronale Netze (JP)
8. 28.04.2015: Support Vector Maschinen I (JP)
9. 05.05.2015: Genetische Fuzzy Systeme (CF)
10. 12.05.2015: Test (3+4+9), Meta-Heuristiken (ACO, PSO) (CF)
11. 19.05.2015: Support Vector Maschinen II (JP)
12. 26.05.2015: Simulated Annealing und andere Suchmethoden (CF)
13. 02.06.2015: Test (7+8+11), Clustering (JP)
14. 09.06.2015: Lernen und Spieltheorie (JP)
15. 23.06.2015: 1. Termin mündliche Prüfungen
16. 30.06.2015: 2. Termin mündliche Prüfungen

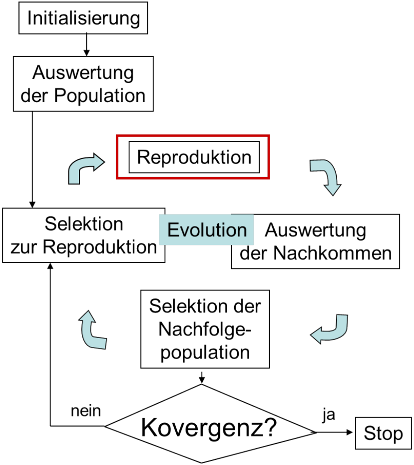
Bemerkungen zum Test nächste Woche

- Schriftlicher Test von ca. 30 min Länge
- Erlaubte Hilfsmittel: Vorlesungsfolien + eigene Mitschriften, Taschenrechner
- **Nicht** erlaubte Hilfsmittel: Laptop, Natel

Lernziele der heutigen Vorlesung

- Erlernen der restlichen Operatoren für Evolutionäre Algorithmen
- Ausführung eines ersten Genetischen Algorithmus
- Erlernen der Funktionsweise von Evolutionsstrategien
- Erlernen der Selbstadaptionsmechanismen für Evolutionsstrategien

Allgemeiner Ablauf



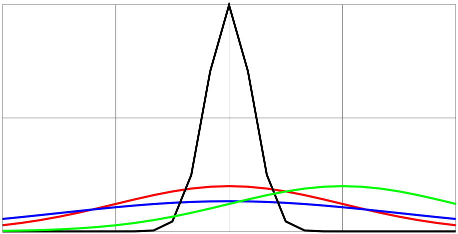
- **Initialisierung**
  - Kodierung
- **Evaluation**
  - Fitnesswert
- **Reproduktion**
  - Rekombination, Mutation
- **Konvergenz**
  - z.B. fixe Generationsanzahl

Reelwertige Mutation

Mit Wahrscheinlichkeit p_m wird ein Objektparametervektor einer Mutation unterzogen.

- Neuer Objektparametervektor für das k-te Individuum: $\tilde{\vec{o}}_k = \vec{o}_k + \vec{z}$
- Es wird angenommen, dass $\vec{o}_k$ und $\vec{z}$ jeweils u Elemente hat.
- Es werden zwei verschiedene Verfahren unterschieden:
  - Isotropische Mutation: $\vec{z} = \sigma(\mathcal{N}_1(0,1), \dots, \mathcal{N}_u(0,1))$
  - Nicht-isotropische Mutation: $\vec{z} = (\sigma_1\mathcal{N}_1(0,1), \dots, \sigma_u\mathcal{N}_u(0,1))$
- $\mathcal{N}(0,1)$ bezeichnet dabei unabhängige Samples der Normalverteilung.
- Damit ergibt sich für den neuen Parametervektor folgende Dichtefunktion: $p(\tilde{o}_k) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(\tilde{o}_k - o_k)^2}{\sigma^2}}$

Normalverteilung



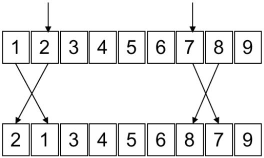
- $\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}}$
- $\mu = 0, \ \sigma = 1$
- $\mu = 0, \ \sigma = 1.5$
- $\mu = 0, \ \sigma = 0.2$
- $\mu = 1, \ \sigma = 1$
- Integral über den ganzen Wertebereich ist stets 1

Reelwertige Mutation

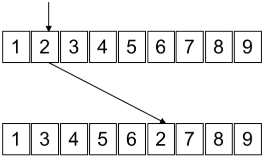
Polynomielle Mutation:

- Neuer Objektparametervektor: $\tilde{o}_k = o_k + \left(o_k^{(U)} - o_k^{(L)}\right)\delta_k$
- Wähle $r_k \in [0,1]$ und $\eta_m \geq 0$
- Bestimme $\delta_k = \begin{cases} (2r_k)^{1/(\eta_m+1)} - 1, & \text{if } r_k < 0.5 \\ 1 - [2(1-r_k)]^{1/(\eta_m+1)}, & \text{if } r_k \geq 0.5 \end{cases}$

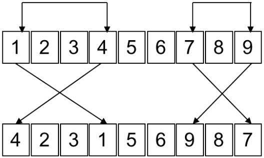
Mutationen bei Sequenzen



Move



Jump



Swap

Mutationen bei Sequenzen

- Probleme dabei sind die noch nicht bekannten Anzahlen der Mutationen und die jeweiligen Distanzen.
- Eingabe
  - Durchschnittliche Mutationsanzahl: *DMZ*
  - Durchschnittliche Distanz: *DD*
- Ausgabe
  - Mutationsanzahl: *MZ*
  - Distanz: *D*
- Berechnung
  - $MZ = \left\lfloor \frac{ld(1-\varsigma_1)}{ld(1-p_{MZ})} \right\rfloor$, mit$p_{MZ} = 1 - \frac{DMZ}{1+\sqrt{1+DMZ^2}}$
  - $D = \left\lfloor \frac{ld(1-\varsigma_2)}{ld(1-p_D)} \right\rfloor$, mit$p_D = 1 - \frac{DD}{1+\sqrt{1+DD^2}}$

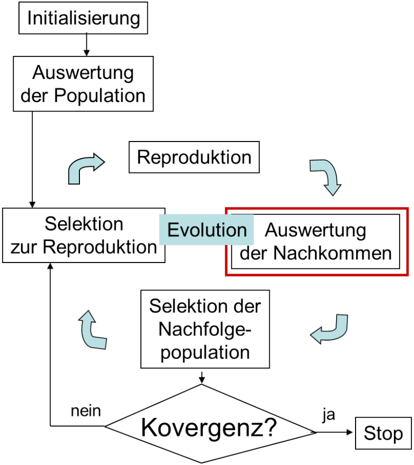
Implementierung

Übung Implementierung

Implementieren Sie bitte:

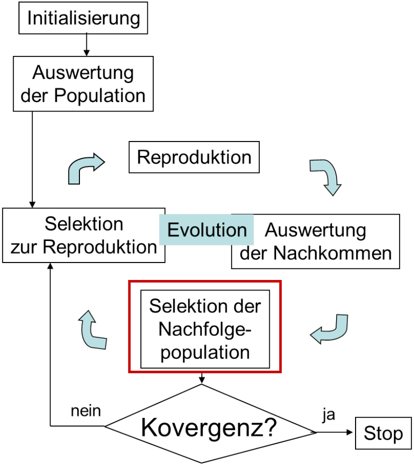
- 1 Funktion, die an einem gegebenen binären String eine Mutation mit Wahrscheinlichkeit p_m vornimmt.
- Wenden Sie diese Funktion auf alle lokalen Individuen an. Das Resultat bildet die Menge der Nachkommen.

Auswertung der Nachkommen



- **Initialisierung**
  - Kodierung
- **Evaluation**
  - Fitnesswert
- **Reproduktion**
  - Rekombination, Mutation
- **Konvergenz**
  - z.B. fixe Generationsanzahl

Auswertung der Nachkommen



- **Initialisierung**
  - Kodierung
- **Evaluation**
  - Fitnesswert
- **Reproduktion**
  - Rekombination, Mutation
- **Konvergenz**
  - z.B. fixe Generationsanzahl

Selektion der Nachfolgegeneration

- Genetisch Algorithmen
  - Es werden nur die Nachkommen für die nächste Generation genutzt. Die Nachkommenanzahl wurde bei der Rekombination beachtet. $(\mu = \lambda)$
- Evolutionsstrategien
  - Verfahrensweise gemäss: $(\mu, \kappa, \lambda, \rho)$

Übung

Übung

Beschreiben Sie mit eigenen Worten die Parameter $(\mu, \kappa, \lambda, \rho)$!

Selektion bei Evolutionsstrategien

- μ: Populationsgrösse der Eltern
- λ: Populationsgrösse der Nachkommen ($\lambda \geq \mu$)
- κ: maximale Anzahl von Lebenszyklen pro Individuum
- ρ: Anzahl der Eltern bei der Rekombination

Algorithmus:

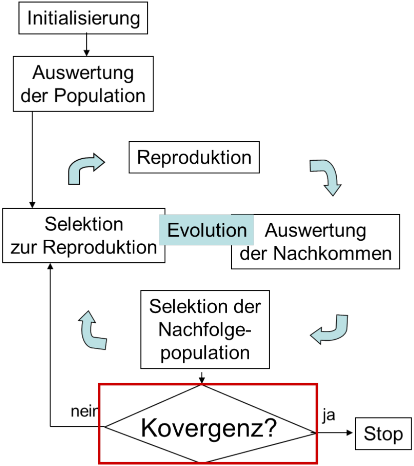
if $(\kappa = 1)$ **then**
  $P_{(t+1),\mu} \leftarrow$ Selektion $(P'_{t,\lambda}, \mu)$;{nicht - elitär}
else if $(\kappa = \infty)$ **then**
  $P_{(t+1),\mu} \leftarrow$ Selektion $(P'_{t,\lambda} \cup P_{t,\mu}, \mu)$;{elitär}
else
  $P_{(t+1),\mu} \leftarrow$ Selektion $(P'_{t,\lambda} \cup P_{t,\mu}, \mu, \kappa)$;{nicht - elitär}
end if

Übung

Übung

Welche Verfahren könnten für die Selektion genutzt werden? Sind die jeweiligen Verfahren dann elitär oder nicht?

Konvergenz



- **Initialisierung**
  - Kodierung
- **Evaluation**
  - Fitnesswert
- **Reproduktion**
  - Rekombination, Mutation
- **Konvergenz**
  - z.B. fixe Generationsanzahl

Konvergenz / Stopp-Kriterien

- Es gibt eine im Prinzip unendliche Anzahl möglicher Kriterien. Gebräuchlich sind unter anderem:
  - Fixe Generationenanzahl
  - Unterschreitung eines Deltas bzgl. der Verbesserung der Fitnessfunktion innerhalb einer gewissen Anzahl von Generationen

Übung

Implementieren Sie einen GA für das bekannte Zylinderproblem

- Initialisierung von 30 Individuen zufällig (binär)
- Auswertung aller 30 Individuen
- For i=1 to 100 //Generationen
  - Rang-basierte Selektion (Individuen müssen Nebenbedingungen erfüllen)
  - Mutation mit pm=10% für die selektierte Menge
  - keine Rekombination
  - Auswertung der Individuen
  - Speicherung des besten Individuums bzgl. f(x) in einem separaten Bereich (für jede Generation ein Individuum)
- Darstellung des Werteverlaufs des jeweils besten Individuums über die 100 Generationen. Was ist die beste gefundene Lösung in (d,h) und welchen Funktionswert hat diese?

Übung

Übung

- Wiederholen Sie den vorherigen Algorithmus für $p_m = \{1\%, 0.5\%, 2\%, 10\%, 20\%\}$.
- Lassen sich Aussagen bezüglich p_m treffen? Wie gross sollte dieser Wert gewählt werden?

Übung

Implementieren Sie einen GA für das bekannte Zylinderproblem

- Initialisierung von 30 Individuen zufällig (binär)
- Auswertung aller 30 Individuen
- For i=1 to 100 //Generationen
  - Rang-basierte Selektion (Individuen müssen Nebenbedingungen erfüllen)
  - Rekombination von 10 zufällig ausgewählten Individuen mittels Single-Point Crossover. Die Eltern werden hier durch die Nachkommen ersetzt.
  - Nutzen Sie die Rekombination, um die Anzahl der Nachkommen mit 30 Individuen sicherzustellen
  - Mutation mit pm=1% für die resultierende Menge
  - Auswertung der Individuen
  - Speicherung des besten Individuums bzgl. f(x) in einem separaten Bereich (für jede Generation ein Individuum)
- Darstellung des Werteverlaufs des jeweils besten Individuums über die 100 Generationen. Was ist die beste gefundene Lösung in (d,h) und welchen Funktionswert hat diese?

Rahmenparameter

- Objektparameter (Problemkodierung): $\vec{o_k}$
- Strategieparameter (Mutation/ Rekombination/ Selektion): $\vec{s_k}$
- Fitness: $F(\vec{o_k})$
- Populationsgrösse zum Zeitpunkt t: $\mu = |P_{t,\mu}|$
- Anzahl Nachkommen: λ
- Anzahl Eltern bei der Rekombination: ρ
- Maximale Anzahl von Generationen pro Individuum: κ

Parameter	**Genetische Algorithmen**	**Evolutionsstrategien**
Individuen	$a_k = (\vec{o_k}, F(\vec{o_k}))$	$a_k = (\vec{o_k}, \vec{s_k}, F(\vec{o_k}))$
Populationsgrösse / Nachkommen	$\mu = \lambda$	$\mu/\lambda = 1/7$
max Anzahl Generationen pro Individuum	$\kappa = 1$	$\kappa \in [1, \infty[$
Strategieparameter	fix	selbstanpassend

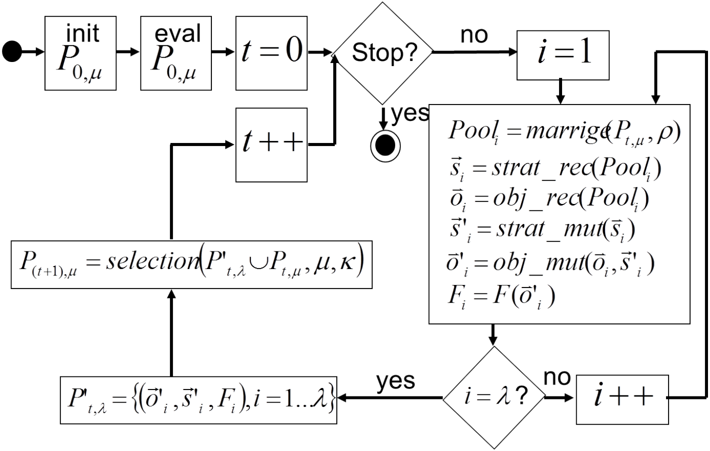
Übung

Übung

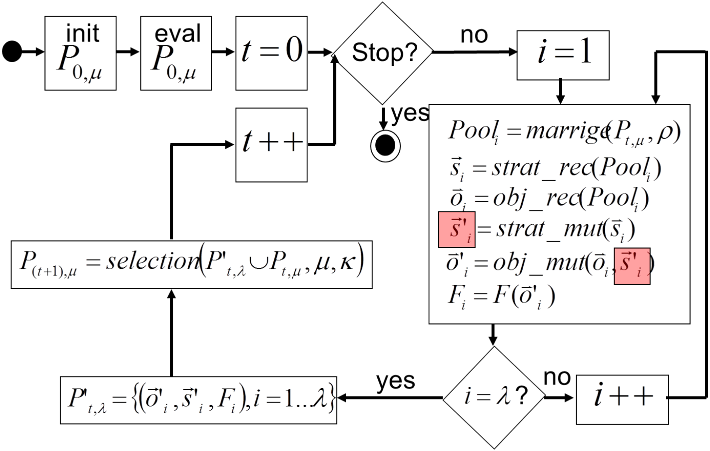
- Beschreiben Sie **eine** mögliche **Selektion der Nachfolgepopulation** für einen Genetischen Algorithmus!
- Beschreiben Sie **eine** mögliche **Selektion der Nachfolgepopulation** für eine Evolutionsstrategie!

- **Exogone** Strategieparameter
  - konstant während der gesamten Optimierung
  - z.B. $(\mu, \kappa, \lambda, \rho)$
- **Endogene** Strategieparameter
  - passen sich während der Optimierung an
  - Beispiele
    - Mutations- / und Rekombinationswahrscheinlichkeiten
    - Schrittweitenparameter bei einzelnen Operatoren

Ablauf einer $(\mu, \kappa, \lambda, \rho)$ Evolutionsstrategie



Ablauf einer $(\mu, \kappa, \lambda, \rho)$ Evolutionsstrategie



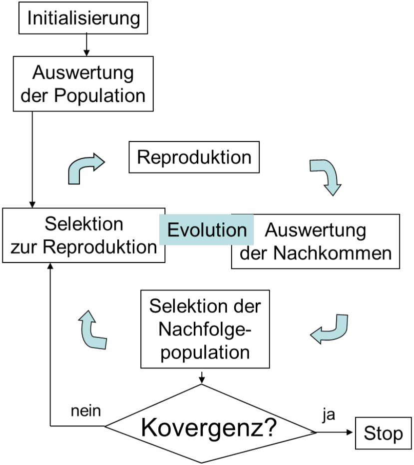
Ablauf einer $(\mu, \kappa, \lambda, \rho)$ Evolutionsstrategie

$P_{0,\mu} \leftarrow$ initialization;
$P_{0,\mu} \leftarrow$ evaluation;
$t \leftarrow 0$;
while (termination criterion not fulfilled) **do**
  for ($i = 1$ to λ) **do**
    $Pool_i \leftarrow$ marriage $(P_{t,\mu}, \rho)$;
    $\vec{s}_i \leftarrow$ strategy_recombination $(Pool_i)$;
    $\vec{o}_i \leftarrow$ object_recombination $(Pool_i)$;
    $\tilde{\vec{s}}_i \leftarrow$ strategy_mutation $(\vec{s}_i)$;
    $\tilde{\vec{o}}_i \leftarrow$ object_mutation $(\tilde{\vec{s}}_i, \vec{o}_i)$;
    $\tilde{F}_i \leftarrow F(\tilde{\vec{o}})$
  end for
  Teil 2
  $t \leftarrow t + 1$
end while

Ablauf einer $(\mu, \kappa, \lambda, \rho)$ Evolutionsstrategie

Teil 1
while (termination criterion not fulfilled) **do**
  Teil 1
  $P'_{t,\lambda} \leftarrow \left\{ \left(\tilde{\vec{o}}_i, \tilde{\vec{s}}_i, \tilde{F}_i \right), i = 1, \ldots, \lambda \right\};$
  if $(\kappa = 1)$ **then**
    $P_{(t+1),\mu} \leftarrow$ selection $(P'_{t,\lambda}, \mu)$;
  else if $(\kappa = \infty)$ **then**
    $P_{(t+1),\mu} \leftarrow$ selection $(P'_{t,\lambda} \cup P_{t,\mu}, \mu)$;
  else
    $P_{(t+1),\mu} \leftarrow$ selection $(P'_{t,\lambda} \cup P_{t,\mu}, \mu, \kappa)$;
  end if
  $t \leftarrow t + 1$
end while

Übung



- Wo finden Sie entsprechende Elemente in der Beschreibung der Evolutionsstrategie?

Selbstanpassung

Der Erfolg einer ES hängt stark von der Selbstanpassung der Strategieparameter ab, die auf veränderte Suchräume reagiert.

- zum Beispiel Mutation: $\tilde{\vec{o}}_k = \vec{o}_k + \vec{z}$
- Es wird angenommen, dass $\vec{o}_k$ und $\vec{z}$ jeweils u Elemente haben.
- Nicht-isotropische Mutation: $\vec{z} = (\sigma_1 \mathcal{N}_1(0,1), \ldots, \sigma_u \mathcal{N}_u(0,1))$
- Selbstadaption verändert nun $\sigma_1, \ldots, \sigma_u$

1/5 Regel (original nur für $(\mu = 1, \lambda = 1, \kappa = \infty)$ Strategien)

$$P_s = \frac{\text{Anzahl erfolgreicher Mutationen}}{\text{Anzahl aller Mutationen}}$$

$$\tilde{\sigma} = \begin{cases} \sigma / a, & \text{falls } P_s > 1/5 \\ \sigma \cdot a, & \text{falls } P_s < 1/5 \\ \sigma, & \text{falls } P_s = 1/5 \end{cases}$$

Der Parameter a wird folgendermassen gewählt: $0,85 \leq a < 1$

Übung

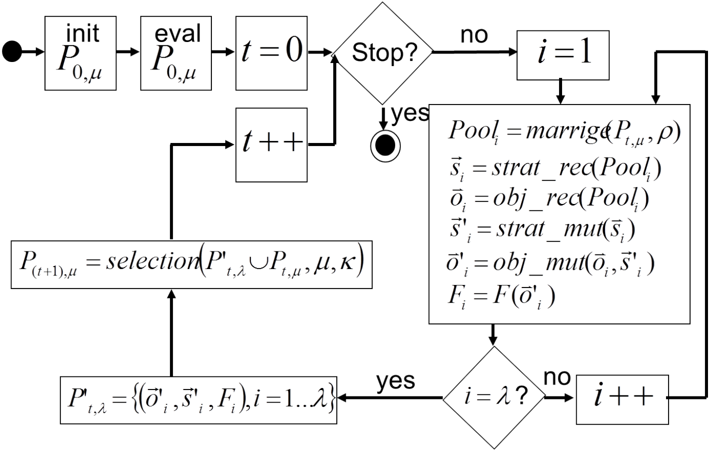
Nehmen Sie die Nutzung der 1/5 Regel an. Welche Aussage(n) ist/sind richtig?

- Wenn 1/5 der letzten Mutationen erfolgreich war, dann wir danach eher lokal gesucht.
- Wenn 1/5 der letzten Mutationen erfolgreich war, dann wird danach eher global gesucht.
- Bei vielen erfolglosen Mutationen wird das Suchgebiet vergrössert um bessere Lösungen zu finden.

- Isotropische Mutation ($\vec{z} = \sigma(\mathcal{N}_1(0,1), \ldots, \mathcal{N}_u(0,1))$):
  - $\tilde{\sigma} = \sigma \cdot e^{\tau \mathcal{N}(0,1)}$
  - Lernrate: $\tau \propto \frac{1}{\sqrt{u}}$
  - Lernrate bei vielen Objektparametern: $\tau \propto \frac{1}{\sqrt{2 \cdot u}}$
- Nicht-isotropische Mutation ($\vec{z} = (\sigma_1 \mathcal{N}_1(0,1), \ldots, \sigma_u \mathcal{N}_u(0,1))$):
  - $\tilde{\vec{\sigma}} = e^{\tau_0 \mathcal{N}(0,1)} \cdot \left(\sigma_1 \cdot e^{\tau_1 \mathcal{N}_1(0,1)}, \ldots, \sigma_u \cdot e^{\tau_1 \mathcal{N}_u(0,1)}\right)$
  - Lernraten:

$$\tau_0 = \frac{1}{\sqrt{2 \cdot u}}, \text{ and } \tau_1 = \frac{1}{\sqrt{2\sqrt{u}}}$$

Wo erfolgt die Selbstanpassung?



Übung

Übung

- Implementieren Sie das bekannte Zylinderproblem mittels einer 100-Generationen Evolutionsstrategie $(\mu, \kappa, \lambda, \rho) = (7, 15, 49, 3)$
- Objektparameter
  - Durschnittliche Rekombination
  - Isotropic Mutation (Start 1%) mit $\tau = \frac{1}{\sqrt{u}}$
- Strategieparameter
  - Diskrete Rekombination
  - Non-Isotropic Mutation (Start 1%) mit

$$\tau_0 = \frac{1}{\sqrt{2 \cdot u}}, \text{ and } \tau_1 = \frac{1}{\sqrt{2\sqrt{u}}}$$

- Zeigen Sie den Verlauf des besten Individuums pro Generation.

Allgemeine Hinweise

Wann sollte was wie genutzt werden?

- Genetische Algorithmen: für diskrete Suchräume
- Evolutionsstrategien: für kontinuierliche Suchräume
  - Falls ES in diskreten Suchräumen genutzt wird: $(\mu, \lambda, \kappa = 1, \rho)$
- **NICHT** als reines Black-Box Optimierungstool
  - Problemkodierung ist sehr wichtig
  - Mutation/Rekombination/Selektion Methoden
  - Methodenparameter, z.B. Mutationsstärke, Rekombinationswahrscheinlichkeiten
- Aber im Allgemeinen sehr mächtig und flexibel!

Hausaufgaben

Hausaufgaben

- Beenden Sie **alle** Beispiele, die Sie während der Vorlesung nicht fertig gestellt haben!
- Dies ist sehr wichtig, da die nachfolgende Vorlesung darauf aufbauen wird.
- Überprüfung zu Beginn der nächsten Vorlesung.
- Durcharbeiten des Papers „Evolution Strategies: A comprehensive introduction". Lesen Sie dabei die Kapitel: 3.1, 3.2, 3.3 (erster Teil), 3.3.2.1, 3.4.1, 4.1.2, 4.2.2.1 und 4.2.2.2.

Vorlesungsplanung - Termine (Änderungen vorbehalten)

1. 17.02.2015: Einkriterielle Evolutionäre Optimierung I (CF)
2. 24.02.2015: Einkriterielle Evolutionäre Optimierung II (CF)
3. 03.03.2015: Test (1+2), Mehrkriterielle Evolutionäre Optimierung I (CF)
4. 10.03.2015: Mehrkriterielle Evolutionäre Optimierung II (CF)
5. 17.03.2015: Statistische Lerntheorie I (JP)
6. 24.03.2015: Statistische Lerntheorie II (JP)
7. 21.04.2015: Test (5+6), Neuronale Netze (JP)
8. 28.04.2015: Support Vector Maschinen I (JP)
9. 05.05.2015: Genetische Fuzzy Systeme (CF)
10. 12.05.2015: Test (3+4+9), Meta-Heuristiken (ACO, PSO) (CF)
11. 19.05.2015: Support Vector Maschinen II (JP)
12. 26.05.2015: Simulated Annealing und andere Suchmethoden (CF)
13. 02.06.2015: Test (7+8+11), Clustering (JP)
14. 09.06.2015: Lernen und Spieltheorie (JP)
15. 23.06.2015: 1. Termin mündliche Prüfungen
16. 30.06.2015: 2. Termin mündliche Prüfungen