

1 Gradientenverfahren

Ausgehend vom Anfangspunkt $\mathbf{x}_0 = [x_1, \dots, x_n]$, berechne Gradient der Kostenfunktion an dieser Stelle: $\nabla c(\mathbf{x}_0) = [\partial c / \partial x_1, \dots, \partial c / \partial x_n]$. Der Nachfolger ist dann, $\mathbf{x}_{i+1} = \mathbf{x}_i + h * \nabla c(\mathbf{x}_i)$, wobei $h > 0$, falls maximiert werden soll und $h < 0$, wenn minimiert wird.

2 Suchverfahren

Suchbaum/Graph

3 stochastische verfahren

3.1 Allgemein

startzustand \mathbf{q} -> einen nachbar \mathbf{p} zufällig wählen -> schritt von ausgang zum neuen nachbar akzeptabel?, falls nein neuer nachbar, sonst von akzeptiertem zustand weiter, bis zurfrieden

3.2 simuliertes tempem

Hier: Maximalzahl betrachteter nachbarn insgesamt (Bedingung fuer Terminieren), Folge sinkender temperaturen t_1, t_2, \dots , Akzeptanzbedingung:

$c(\mathbf{p}) > c(\mathbf{q})$ OR $\exp(\frac{c(\mathbf{p}) - c(\mathbf{q})}{t_i}) > Zufallszahl \in [0, 1]$, nächste Temperatur immer wenn akzeptiert wurde.

3.3 Schwellwert-Algorithmus

Akzeptanzbedingung: $c(\mathbf{p}) > c(\mathbf{q}) - \sigma$, wobei σ immer weiter abgesenkt wird.

3.4 Sintflut-Maximierung

Akzeptanzbedingung: $c(\mathbf{p}) > F$, mit steigender unterer grenze F.

3.5 Rekordjagd-Algorithmus

Akzeptanzbedingung: $c(\mathbf{p}) \geq \text{Rekord} - \sigma$, bisher bester gesehener Wert, darf nicht um sinkende Toleranz unterschritten werden.

4 Evolutionäre Algorithmen

4.1 Allgemein

Population aus Individuen mit Merkmalsvektor \mathbf{q} und Fitness $c(\mathbf{q})$. Außerdem gibt es noch einen globalen Streuvingsvektor σ , aus dem normal oder geometrisch verteilt mutiert

wird. Individuen koennen sich fortpflanzen, bei mehreren Eltern Kreuzung, sonst Klon. Populationsgröße wird einigermaßen konstant gehalten, d.h. es werden immer wieder Loesungen verworfen.

4.2 Plus-Evolutionsstrategie

Erzeuge λ Nachkommen, nur die μ fittesten Individuen aus den μ Eltern und λ Nachkommen ueberleben.

Eltern werden pro Kind zufällig aus Population gewählt.

4.3 Komma-ES

Wie bei plus, allerdings sterben alle Eltern garantiert und nur μ fittesten Kinder ueberleben.

4.4 Klonen vs. Mehrere Eltern

Beim Klonen mutiere einfach mithilfe von σ (s.o.).

Bei mehreren Eltern:

- Mischen: Es wird fuer jedes Merkmal (q_i) zufaellig bestimmt, von welchem Elternteil dieses Merkmal uebernommen wird.
- Mitteln: Es wird fuer jedes Merkmal der Mittelwert ueber die Werte der Eltern gebildet.

5 Genetische Algorithmen

Binäre Merkmalsvektoren, nur Nachkommen ueberleben.

Individuum klont sich mit Wahrscheinlichkeit $W(\mathbf{q}) = c(\mathbf{q}) / \sum_{\mathbf{p} \in Pop} c(\mathbf{p})$. μ Eltern erzeugen immer μ Klon-Nachkommen.

Wähle unter den μ Klonen $p\%$ Individuen, die gekreuzt werden. Wähle daraus zufällige Paare. Aus $a = (a_1, \dots, a_n)$ und $b = (b_1, \dots, b_n)$ entstehen $c = (a_1, \dots, a_j, b_{j+1}, \dots, b_n)$ und $d = (b_1, \dots, b_j, a_{j+1}, \dots, a_n)$.

Dann kippe jedes Bit des Merkmalsvektors mit sehr geringer Wahrscheinlichkeit.

Zusammengehörende Merkmalsbits sollten möglichst nahe beisammen stehen, damit sie bei der Kreuzung nicht zerbrechen.

6 Partikel-Schwarm-Optimierung

Partikel jeweils mit Position $p_i(t)$, Fitness $f_i(p_i)$, Geschwindigkeit $v_i(t) = p_i(t) - p_i(t-1)$. Algorithmus:

Initiale Position und Geschwindigkeit zufällig. Dann sei q_i die bisher beste Position eines Partikels, q die global bisher beste Position.

Dann nächste Werte:

$v_i = v_i * \omega + (q_i - p_i) * c_1 * r_1 + (q - p_i) * c_2 * r_2$ r jeweils aus $[0,1]$, c und ω konstant.
Die Konstanten können per Superschwarm optimiert werden.

7 Ameisen-Systeme

Formeln abschreiben