

Metodi Computazionali della Fisica

Secondo Modulo: C++

Sesta Lezione



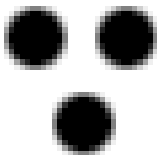
Ricostruzione di una immagine

Proposte per il progetto personale

Esame

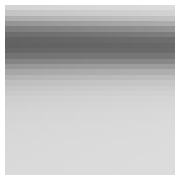
Ultimi ritocchi

Un semplice esempio



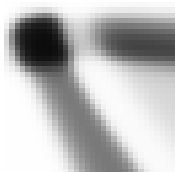
- ▶ immagine pgm, 32x32;
- ▶ mlp.par: 2-9-7-1, funzione di attivazione sigmoide su tutti i livelli;
- ▶ ga.par: 1 clone, 100 mutanti, rate 1.0, 2 mutazioni multiple;
- ▶ minimization.par: 10000 iterazioni, 20% dei punti nel set di training.

Un semplice esempio



- ▶ immagine pgm, 32x32;
- ▶ mlp.par: 2-9-7-1, funzione di attivazione sigmoide su tutti i livelli;
- ▶ ga.par: 1 clone, 100 mutanti, rate 1.0, 2 mutazioni multiple;
- ▶ minimization.par: 10000 iterazioni, 20% dei punti nel set di training.

Un semplice esempio



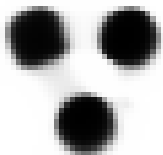
- ▶ immagine pgm, 32x32;
- ▶ mlp.par: 2-9-7-1, funzione di attivazione sigmoide su tutti i livelli;
- ▶ ga.par: 1 clone, 100 mutanti, rate 1.0, 2 mutazioni multiple;
- ▶ minimization.par: 10000 iterazioni, 20% dei punti nel set di training.

Un semplice esempio



- ▶ immagine pgm, 32x32;
- ▶ mlp.par: 2-9-7-1, funzione di attivazione sigmoide su tutti i livelli;
- ▶ ga.par: 1 clone, 100 mutanti, rate 1.0, 2 mutazioni multiple;
- ▶ minimization.par: 10000 iterazioni, 20% dei punti nel set di training.

Un semplice esempio



- ▶ immagine pgm, 32x32;
- ▶ mlp.par: 2-9-7-1, funzione di attivazione sigmoide su tutti i livelli;
- ▶ ga.par: 1 clone, 100 mutanti, rate 1.0, 2 mutazioni multiple;
- ▶ minimization.par: 10000 iterazioni, 20% dei punti nel set di training.

Radial Basis Functions

Da fare:

1. implementare l'algoritmo della RBF nella classe `Parametrization`;
2. effettuare il confronto con le altre parametrizzazioni;
3. implementare l'algoritmo di inversione di una matrice nella classe `Minimization`;
4. confrontare le prestazioni nella ricostruzione di una semplice immagine.

Radial Basis Functions

Da fare:

1. implementare l'algoritmo della RBF nella classe `Parametrization`;
2. effettuare il confronto con le altre parametrizzazioni;
3. implementare l'algoritmo di inversione di una matrice nella classe `Minimization`;
4. confrontare le prestazioni nella ricostruzione di una semplice immagine.

(Steepest) Gradient Descent - Conjugate Gradient

Da fare:

1. implementare l'algoritmo per il calcolo di una derivata nella classe `Minimization`;
2. implementare l'algoritmo SD e/o CG nella classe `Minimization`;
3. confrontare le prestazioni delle diverse minimizzazioni nella ricostruzione di dati artificiali generati con una funzione nota;
4. confrontare le prestazioni dei diversi algoritmi applicati ad una parametrizzazione polinomiale e ad una basata su rete neurale (MLP).

Tabu Search

Da fare:

1. implementare l'algoritmo del TS nella classe `Minimization`;
2. confrontare le prestazioni con l'algoritmo genetico nella ricostruzione di dati artificiali generati con una funzione nota e nella ricostruzione di una semplice immagine.

“Google - strumenti per le lingue”

Da fare:

1. generare una serie semplici frasi (ad esempio, tre parole) in due lingue usando il traduttore di Google;
2. implementare la conversione da caratteri/stringhe a numeri;
3. definire una funzione di merito per il fit;
4. *addestrare* la rete neurale sui dati generati;
5. controllare il risultato con frasi non utilizzate nel fit.

Filtro anti-Spam

Da fare:

1. generare una serie semplici *subject*;
2. implementare la conversione da caratteri/stringhe a numeri;
3. definire una funzione di merito per il fit;
4. *addestrare* la rete neurale sui dati generati;
5. controllare il risultato con frasi non utilizzate nel fit.

Altro

- ▶ Un convertitore di formati.
- ▶ Un codice di compressione dati (ad esempio, LZW).
- ▶ "Il commesso viaggiatore".
- ▶ "Il paroliere".
- ▶ Varie ed eventuali.

Criteri di valutazione

- ▶ relazione;
- ▶ chiarezza del codice;
- ▶ stabilità ed efficienza del codice;
- ▶ presentazione e discussione del progetto;
- ▶ completezza strumenti utilizzati;
- ▶ difficoltà di progettazione.

Note:

- ▶ il progetto può essere svolto a coppie purché il contributo di ognuno sia chiaramente riconoscibile e valutabile;
- ▶ la discussione del progetto è prevista per il giorno 31 gennaio 2008 alle ore 14.00.

Test e debug del codice

1. Controllate che la classe `Minimization` compili;
2. implementate almeno uno dei metodi di dump suggeriti;
3. controllate che il programma funzioni e faccia quello vi aspettate;
4. provate a variare alcuni dei parametri della parametrizzazione e/o del fit per avere un'idea del loro effetto.

Strumenti per il debug

- ▶ `ddd` (GUI di `gdb`);
- ▶ `valgrind`;
- ▶ `strace`.