

Blatt 2

Abgabe bis Montag, 26. April 2010, 13:00 Uhr bei Dr. Konrad Rieck per email
 (konrad.rieck@tu-berlin.de)

Auf diesem Übungszettel soll t-SNE praktisch untersucht werden. Eine entsprechende Beispielimplementation in Matlab befindet sich hier:

http://ticc.uvt.nl/~lvdrmaaten/Laurens_van_der_Maaten/t-SNE.html

Die entsprechende Implementation heisst "Simple Matlab implementation". Außerdem werden die Daten `mnist_train.mat` von der Vorlesungswikiseite benötigt. Interessante Informationen finden sich auch im "User's Guide".

Zur Information: Eine Lehreinstallation von Matlab findet Ihr auf den Rechnern des IRB (fiesta, pepino, usw.) unter

`/home/ml/ml/bin/matlab`

Die praktische Aufgabe besteht darin, die fehlenden Funktionen im Programmskelett zu ergänzen:

1. **swissroll (10 Punkte)** Diese Funktion erzeugt den sogenannten Swissroll-Datensatz:

$$\begin{aligned} Y_i &\sim \text{uniform in } [0, 2\pi] \\ X_i &= ((1 + Y_i) \cos(Y_i), (1 + Y_i) \sin(Y_i), \varepsilon_i) \\ \varepsilon_i &\sim \text{uniform in } [0, 1]. \end{aligned}$$

Wie schätzt Du die Qualität der Lösung ein? Versuche verschiedene Parameterwerte, um die Darstellung zu verbessern.

2. **show_mnist (20 Punkte)** Diese Funktion plottet eine zufällige Auswahl von 50 Beispielen der niedrigdimensionalen mnist-Daten. Hierbei sollte wie folgt vorgegangen werden:

- (a) Plote alle Datenpunkte als Punkte.
- (b) Wähle zufällig 50 Datenpunkte aus.
- (c) Plote an den entsprechenden Positionen die eigentlichen Ziffern (siehe unter anderem `axes`, `pcolor`, und `axis off`, `shading flat`)

Hinweis: Da die Koordinaten von `axes` relativ zum gesamten Fenster sind, müssen die Datenpunkte im ersten Schritt ebenfalls "von Hand" umgerechnet werden.

Hinweis: Zum Debuggen empfiehlt es sich, gegebenenfalls die Anzahl der Daten oder die maximale Anzahl von Iterationen (Variable `max_iter` in `tsne.m`) herunterzusetzen, oder Zwischenergebnisse abzuspeichern.

```
function sheet02

% generate a swissroll data set
[X, Y] = swissroll(1000);

figure(1)
scatter3(X(:, 1), X(:, 2), X(:, 3), 30, Y, 'filled')

% apply t-SNE
no_dims = 2;
init_dims = 3;
perplexity = 50;
```

```

%mappedX = tsne(X, [], no_dims, init_dims, perplexity);
mappedX = X

% plot mapped data
figure(2)
scatter(mappedX(:, 1), mappedX(:, 2), 20, Y);

% load mnist data
load mnist_train.mat
ind = randperm(size(train_X, 1));
train_X = train_X(ind(1:1000),:);
train_labels = train_labels(ind(1:1000));

no_dims = 2;
init_dims = 30;
perplexity = 30;

mappedX = tsne(train_X, [], no_dims, init_dims, perplexity);

% plot data
figure(3)
show_mnist(mappedX, train_X, train_labels);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Your solutions below
%

% Generate swissroll data
function [X, Y] = swissroll(N)
% your solution here...

% Show results on the mnist data set. Show all points as dots,
% show 50 random samples as small image.
function show_mnist(mappedX, train_X, train_labels)
% your solution here...

```
