

Exercises to the lecture

Machine learning and Pervasive Computing

Summer term 2015

Stephan Sigg (stephan.sigg@cs.uni-goettingen.de)

Exercise: 18.05.2015, 16:15 – 17:45

Assignment 02: Regression

- a) Implement the gradient descent function to compute optimum regression parameters. You may extend the Octave/Matlab function below for a start. Implementations in other languages are fine too.

```
function [W, E_history] = gradientDescent(X, y, W, delta, num_iter)
%gradientDescent learns W via gradient descent
% W = gradientDescent(X, y, W, delta, num_iter) updates W in
% num_iter steps with learning rate delta

% Initialisation
m = length(y); % # of training examples
E_history = zeros(num_iter, 1);

for iter = 1:num_iter

    % ===== YOUR CODE HERE =====
    % Perform gradient step on the parameter vector W.
    %

    % =====

    % Save the cost E in every iteration
    E_history(iter) = computeCost(X, y, W);

end
end
```

- b) Assuming a trained logistic regression classifier with $w_0 = -2.21$, $w_1 = 4.28$. Classify the samples

x_1		0.8534
x_2		0.5163
x_3		0.764
x_4		0.432
x_5		0.7485
x_6		0.229
x_7		0.964