# Homework 6: logistic regression

## Fundamentals of Data Science 2015

Download the dataset

`http://www.umass.edu/statdata/statdata/data/nhanes.xls`

and save it in CSV format, separating all fields by a comma and keeping a first row containing the column names. You must write a Python script, named `ID.py` where `ID` is your student's ID, which takes the name of the aforementioned CSV file as a command-line parameter and:

1. reads the data, drops the first column (or uses it as an index) and drops the column named "BMI", replaces each missing value with the median value of its column, and normalizes all columns to have zero mean and unitary standard deviation
2. performs a logistic regression to predict the label i.e. the value of the last column (which takes on values in $\{0, 1\}$), as a function of all other columns and of an intercept term (equal for all data points); therefore for a generic point $x^{(i)}$ the predicted label $\hat{y}^{(i)}$ will be the value of the expression "$h_\theta(x^{(i)}) > 0.5$" where $h_\theta(x)$ is the logistic function evaluated at $\theta^T x$, for some $\theta$ to be found (and after adding the intercept term to all $x^{(i)}$).

The script should output:

1. the vector $\hat{\theta}$ that maximizes the log-likelihood of the logistic function given the data
2. the fraction of data points $x^{(i)}$ for which the predicted label $\hat{y}^{(i)}$ is correct

You should find $\hat{\theta}$ using the gradient descent technique applied to the log-likelihood of the logistic function (check the material on the course's website). Note that we want to *maximize* the log-likelihood, and thus the "descent" is actually an *ascent* (i.e. one moves towards the gradient rather than against it). Using fitting functions provided by existing libraries is not allowed. You are free to use other techniques such as Newton-Raphson in place of gradient descent.

As usual, the homework is due before next Thursday.