Machine Learning for Data Science

http://www1.ccls.columbia.edu/~ansaf/4721/
COMS 4721 – Spring 2014

Ansaf Salleb-Aouissi
as2933@columbia.edu

Home Work n°1: Linear Regression (complete)

Preliminaries

1. Part II.2: Released March 1st, 2014. Due March 10, 2014
2. Part I and Part II.1: Released February 16, 2014. ~~Due March 3, 2014~~.

   PLEASE SUBMIT THE COMPLETE HOMEWORK BY MARCH 10 BY 11:59pm.
3. We recommend you follow the order in which the problems are provided.
4. Mandatory: Please comment **all important lines** in your code (except the obvious lines).
5. Please use Python in solving all problems in Section II.
6. You can use any Python package to solve the problems, including **numpy** and **matplotlib**, but do not use **scikit-learn** nor another ready to use implementation of linear regression including **stats.linregress**. The goal of the exercise is that you implement from scratch to get a better understanding of the methodology.
7. Use Latex to compile your answer. Latex is preferred but not mandatory. For your responses addressing the Conceptual questions, submit one PDF document only. Make sure you number all your answers.
8. Your submission package must be a zipped file of: PDF containing all your responses, separate code for each problem, and figures as requested in png or pdf form.
9. Name your submission as follows: firstname_lastname_uni.zip and submit on CourseWorks.
10. Enjoy!

I. Conceptual Questions

Question 1: True or False?

1. Supervised and unsupervised learning both aim to identify *classes* in data **(at the learning stage not prediction)**.
2. When feature space is large, overfitting is likely.
3. Overfitting can be controlled by regularization.
4. Once you learn a classification model, you can use the test set to assess the model performance.
5. If the performance of a classification model on the test set is poor, you can re-calibrate your model parameters to achieve a better model.

6. Cross-validation is used only when one have a large training set.

7. The examples in a validation set are used to train a classification model.

8. To learn a regression model you can either use gradient descent or normal equations.

9. Because it is straightforward to calculate in just one step, Normal equation is the preferred method when the feature space is large (e.g., 10,000 features).

10. If the learning rate α is small enough, gradient descent converges very fast.

11. Ridge regression aims to increase the variance of linear regression by decreasing the bias.

12. Lasso is a variant of linear regression that calculates a sparse solution.

13. K-NN works only for classification.

14. K-NN is a linear classification method.

15. The ℓoss function aggregates the classification/regression error on all examples.

Question 2: Machine Learning Definition in Practice

Recall: "A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E. " Tom Mitchell.

What are the sets E, T, and P in the case of a *Recommender System*? Please justify your answer and elaborate with examples.

II. Practical Problems

---

1 Linear regression with one feature

We are interested in studying the relationship between age and height (statures) in girls aged 2 to 8 years old. We think that this can be modeled with a linear regression model. We have examples (data points) for a population of 60 girls. Each example has one feature *Age* along with a numerical label *Height*. We will use the dataset **girls_train.csv** (derived from CDC growthchart data[1]). Your mission is to implement linear regression with one feature using gradient descent. You will plot the data, regression line, coefficient contours and cost function. You will finally make a prediction for a new example using your regression model and assess your out of sample mean square error.

1.1 Load & Plot

a) Load the dataset **girls.csv**.

b) Plot the distribution of the data. You should be getting a plot similar to Figure 1.

[1] http://www.cdc.gov/growthcharts/

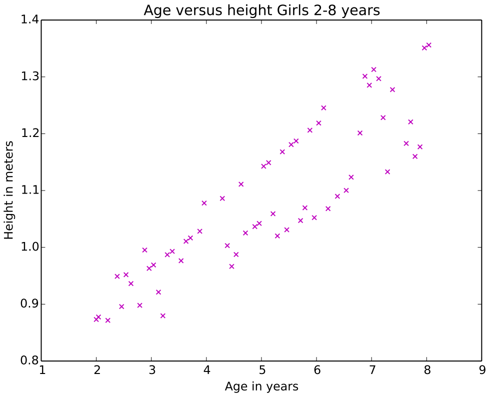

Figure 1: Scatter plot.

1.2 Gradient descent

Now that your data is loaded and you know how it looks like, find a regression model of the form:

$$\text{Height} = \beta_0 + \beta_1 \times \text{Age}$$

a) Implement Gradient Descent to find the β's of the model. Remember you need to add the vector 1 ahead of your data matrix. Also, make sure you update the parameters β's *simultaneously*. Use a learning rate alpha = 0.05 and #iterations = 1500.

b) What is the mean square error of your regression model on the training set?

1.3 Plot the regression line, contours and bowl function

Once you obtain your parameters:

a) Plot the regression line on top of your distribution. You should obtain a plot like Figure 2 (Left).

b) Plot the contour lines for your β's (similar to the one in Figure 2 (Right)).

c) Plot the cost function that has the bowl shape as we saw in the class.

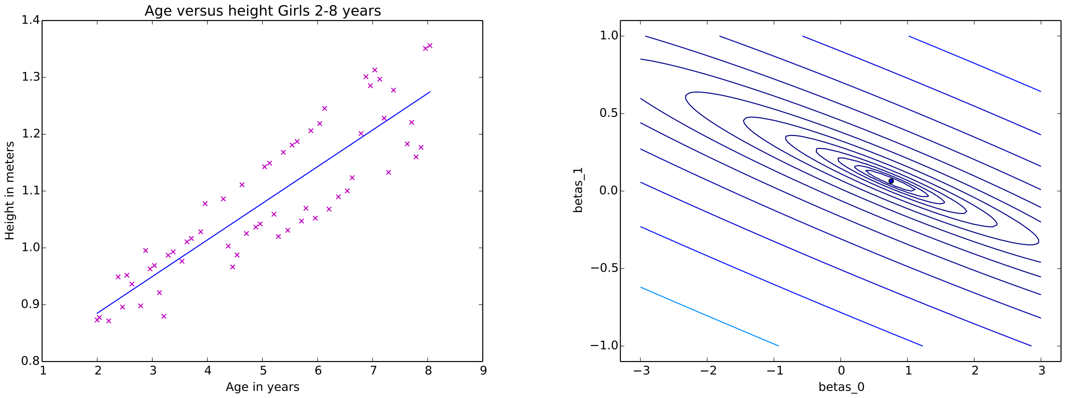

Figure 2: (Left) Regression line. (Right) Contour lines for the β's.

1.4 Testing your model and Making a prediction for a new example

a) It's time now to try your model and make a prediction. Using your model, make a prediction for a 4.5 years old girl. What would be her predicted height?

b) Given a dataset of 20 girls that was kept on the side as a test set **girls_test.csv**, compute the mean square error for the test set and make a comparison to the training mean square error.

2 Linear regression with multiple features

In this problem, you will work on linear regression with multiple features using gradient descent and the normal equation. You will also study the relationship between the risk function, the convergence of gradient descent, and the learning rate. We will use the dataset **girls_age_weight_height_2_8.csv** (derived from CDC growthchart data).

2.1 Data Preparation & Normalization

Once you load your dataset, explore the content to identify each feature. Remember to add the vector 1 (intercept) ahead of your data matrix.

a) You will notice that the features are not on the same scale.They represent age (years), and weight (kilograms). Print the mean and standard deviation of each feature in your data. The last column is the label and represent the height (meters).

b) Scale each feature by its standard deviations and set its mean to zero. You do not need to scale the intercept. For the each feature x (a column in your data matrix), use the following formula:

$$x_{\text{scaled}} = \frac{x - \mu(x)}{stdev(x)}$$

2.2 Gradient Descent

As you did in the previous section, implement gradient descent but this time with two features. Initialize your β's to zero. We recall the empirical risk and gradient descent rule as follows:

$$R(\beta) = \frac{1}{2n}\sum_{i=0}^{n}(f(x_i) - y_i)^2$$

$$\forall j \qquad \beta_j = \beta_j - \alpha\frac{1}{n}\sum_{i=0}^{n}(f(x_i) - y_i)x_i$$

2.3 Plotting Risk function for different learning rates

a) We need to pick an appropriate learning rate $\alpha \in \{0.005, 0.001, 0.05, 0.1, 0.5, 1\}$. Run gradient descent and plot the Risk function with respect to the number of iterations for different values of α. Use the same figure to plot all curves. Choose #iterations=50.

b) Compare the convergence rate when α is small versus large.

c) Which α is best? Use this α to run gradient descent and print the β's.

2.4 Normal equation

Recall the second approach for linear regression using the Normal Equation. Its implementation is straightforward to find the β's. There is no need to scale the features this time.

$$\beta = (X^t X)^{-1} X^T y$$

a) Compare the β vector you obtained with gradient descent to the one calculated with normal equation. Are they the same? Why?

2.5 Prediction

a) Using both β vectors (the one obtained with gradient descent and the one obtained with normal equations), make a height prediction for a 5-year old girl weighting 20 kilos (don't forget to scale!).

b) Do gradient descent and Normal Equation lead to the same height prediction?