Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg

Teacher: J. Hennebert
Assistant: Y. Iseli

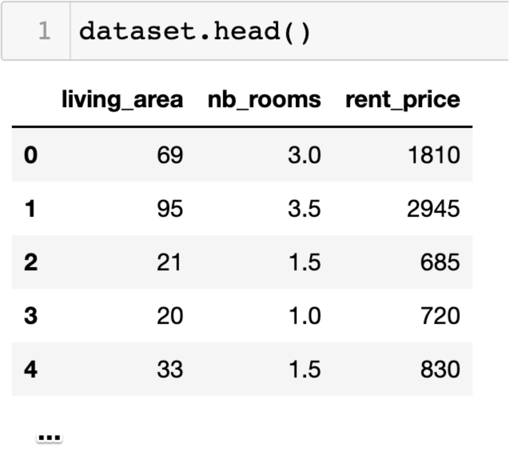
ISC

Intro to Deep Learning

02.03.2023
PW 01 - Part 1 – Linear Regression with Gradient Descent

There is a **part 2** to this practical work. Both parts A and B are to be submitted together, check the submission dates on Moodle.

Start from the Python notebook `pw1-linear-regression-stud.ipynb` and the data provided in `lausanne-appart.csv` available on Moodle. We will play with the appartment renting price in the city of Lausanne, see Figure 1.



```
dataset.head()
```

	living_area	nb_rooms	rent_price
0	69	3.0	1810
1	95	3.5	2945
2	21	1.5	685
3	20	1.0	720
4	33	1.5	830

...

FIGURE 1 – Lausanne appartment data

In the case of a monovariable linear regression, we want to discover the parameters θ of a simple linear model defined with

$$h_\theta(\mathbf{x}) = \theta_0 + \theta_1 x \tag{1}$$

The "best" θ's are the ones that will minimise the squared difference between the gotten outputs and the target outputs, the *MSE loss function* that is defined with

$$J(\theta) = \frac{1}{2N} \sum_{n=1}^{N} (h_\theta(\mathbf{x}_n) - y_n)^2 \tag{2}$$

Exercise 1 Gradient descent for linear regression

Implement the *full batch* gradient descent algorithm for the previous problem. As seen in the theory, the update rules are

$$\theta_0 \leftarrow \theta_0 - \alpha \frac{1}{N} \sum_{n=1}^{N} (h_\theta(\mathbf{x}_n) - y_n) \tag{3}$$

$$\theta_1 \leftarrow \theta_1 - \alpha \frac{1}{N} \sum_{n=1}^{N} (h_\theta(\mathbf{x}_n) - y_n) x_{n,1} \tag{4}$$

Remark You need to iterate several times over the training set. If you have problems of convergence, you need to use a smaller value of α. Values such as 0.000001 are common.

a) Plot the cost value (Equation 2) as a function of the iterations. What do you observe ?

b) Imagine a stopping criterion, i.e. when do we stop iterating on the training set ?

c) Plot the computed line $h_\theta(\mathbf{x})$ on top of the scatter plot of exercise 1.

d) Compute the final cost value according to Equation 2 and compare it to the one of exercise 2. What can you conclude ?

Exercise 2 Stochastic gradient descent for linear regression

Implement the stochastic gradient descent algorithm for the previous problem. As seen in the theory, the update rules are

$$\theta_i \leftarrow \theta_i - \alpha (h_\theta(\mathbf{x}_n) - y_n) x_{n,i} \tag{5}$$

a) Plot the computed line $h_\theta(\mathbf{x})$ on top of the scatter plot of exercise 1.

b) How many samples do you need to visit for reaching the convergence ?

c) What kind of stopping criterion could we use here ?

d) Compute the final cost value according to Equation 2 and compare it to the one of exercise 2 and 3. What can you conclude ?

Exercise 3 Review questions

a) The linear regression has a mathematical *closed form* solution. Then, in which conditions would we prefer a gradient descent algorithm to compute the regression model ?

b) Outliers in a data set can be defined as values that are out of the "usual" range in comparison with other values. They typically come from noise or anomalies in the data capturing process. What is the impact of an outlier in the stochastic gradient descent process ? What if we have many outliers ? *Hint* : look at the equation of the MSE, and to the equation of the update rule.

c) In the case of stochastic gradient descent, what is the danger of having a too large or too small α value ? Could you think of a better (more advanced) strategy as the one stated in Slide 36 ?

d) Let's assume we expect that the target variable y has a dependency to the square and to the cube of one of the feature x_d in our multi-variable training set $(x_1, \ldots, x_d, \ldots, x_D)$. How would you proceed ? Do we need to take precautions in terms of numerical stability ?

e) *Advanced.* Could we use a descent algorithm without computing the gradient ? If yes, give a pseudo code of the algorithm to find (θ_0, θ_1) using a linear regression $h_\theta(x) = \theta_0 + \theta_1 x$.

Exercise 4 Optional – Mini-batch gradient descent for linear regression

Implement the mini-batch gradient descent algorithm for the previous problem, adding a parameter B defining the size of the mini-batch. Check that when $B = N$, you fall back on the batch gradient descent solution, and when $B = 1$, you get the behaviour of stochastic gradient descent.

Exercise 5 Optional – multi-variable linear regression

a) Implement one of the gradient descent algorithm (ex. 3-5) for the multi-variable linear regression assuming x_1 being the living area and x_2 the square of the living area. Plot the computed curve (second order) on top of the scatter plot of exercise 1.

b) Implement one of the gradient descent algorithm (ex. 3-5) for the multi-variable linear regression assuming x_1 being the living area and x_2 the number of bedrooms.