0845209
Siméon Gaumart

Machine Learning HW1
2020.3.23

Problem 1 :

a)
We know that $\hat{y}=\hat{\beta}_0+\hat{\beta}_1 x \Leftrightarrow \hat{\beta}_0=\hat{y}-\hat{\beta}_1 x$ and $y=\beta_0+\beta_1 x+u \Leftrightarrow x=(y-\beta_0-u)/\beta_1$
so $\hat{\beta}_0=\hat{y}-\frac{\hat{\beta}_1}{\beta_1}(y-\beta_0-u) \Leftrightarrow E[\hat{\beta}_0]=E[\hat{y}-\frac{\hat{\beta}_1}{\beta_1}(y-\beta_0-u)]$

with the question b), we will demonstrate that $E[\hat{\beta}_1]=\beta_1$

so supposing it $E[\hat{y}-\frac{\hat{\beta}_1}{\beta_1}(y-\beta_0-u)]=E[\hat{y}-\frac{\beta_1}{\beta_1}(y-\beta_0-u)]=E[\hat{y}-y+\beta_0+u]$

furthermore, $u=y-\hat{y}$

so $E[\hat{y}-y+\beta_0+u]=E[-u+\beta_0+u]=E[\beta_0]=\beta_0$ because it's a constant.

So at the end, $E[\hat{\beta}_0]=\beta_0$

b)
We know that

$$\hat{\beta}_1=\beta_1+\frac{\sum_{i=1}^{n}(x_i-\bar{x})u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2} \Leftrightarrow \hat{\beta}_1=\beta_1+\frac{\sum_{i=1}^{n}x_i u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}-\frac{\sum_{i=1}^{n}\bar{x}u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2} \Leftrightarrow \hat{\beta}_1=\beta_1+\frac{\sum_{i=1}^{n}x_i u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}-\frac{n\bar{x}\sum_{i=1}^{n}u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$$

so

$$E[\hat{\beta}_1]=E[\beta_1+\frac{\sum_{i=1}^{n}x_i u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}-\frac{n\bar{x}\sum_{i=1}^{n}u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}]=E[\beta_1]+E[\frac{\sum_{i=1}^{n}x_i u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}]-E[\frac{n\bar{x}\sum_{i=1}^{n}u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}]$$

And we know that Cov(x,u) = E[xu] = 0

so $E[x_i u_i]=0 \Rightarrow E[\sum_{i=1}^{n}x_i u_i]=0 \Rightarrow E[\frac{\sum_{i=1}^{n}x_i u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}]=0$

and we know too that E[u] = 0 and $E[n\bar{x}]=n\bar{x}$ because it's a constant.

so $E[u_i]=0 \Rightarrow E[\sum_{i=1}^{n} u_i]=0 \Rightarrow E[\frac{n\bar{x}\sum_{i=1}^{n} u_i}{\sum_{i=1}^{n}(x_i-\bar{x})^2}]=0$

and we know that $E[\beta_1]=\beta_1$ because it's a constant.

So at the end, $E[\hat{\beta}_1]=\beta_1$

Problem 2 :

a)
In this problem, we have to estimate $\hat{\beta}_0$ and $\hat{\beta}_1$ in the equation
$\hat{math10}$ = $\hat{\beta}_0$ + $\hat{\beta}_1$ * lnchprg.

Using the data of math10 and lnchprg, we have to do a linear regression.

For this, we have to calculate the loss function $S=\sum_i [math10_i-(\beta_0+lnchprg_i\beta_1)]^2$

and to solve the equation system $\begin{cases} \frac{\delta S}{\delta \beta_0}=0 \\ \frac{\delta S}{\delta \beta_1}=0 \end{cases}$

we have $\begin{cases} \frac{\delta S}{\delta \beta_0}=\sum_i 2[math10_i-(\beta_0+lnchprg_i\beta_1)](-1) \\ \frac{\delta S}{\delta \beta_1}=\sum_i 2[math10_i-(\beta_0+lnchprg_i\beta_1)](-lnchprg_i) \end{cases}$

$\Leftrightarrow \begin{cases} \frac{\delta S}{\delta \beta_0}=(\sum_i 2)\beta_0+(\sum_i 2\,lnchprg_i)\beta_1+(\sum_i(-2\,math10_i)) \\ \frac{\delta S}{\delta \beta_1}=(\sum_i 2\,lnchprg_i)\beta_0+(\sum_i 2\,lnchprg_i^2)\beta_1+(\sum_i(-2\,math10_i\,lnchprg_i)) \end{cases}$

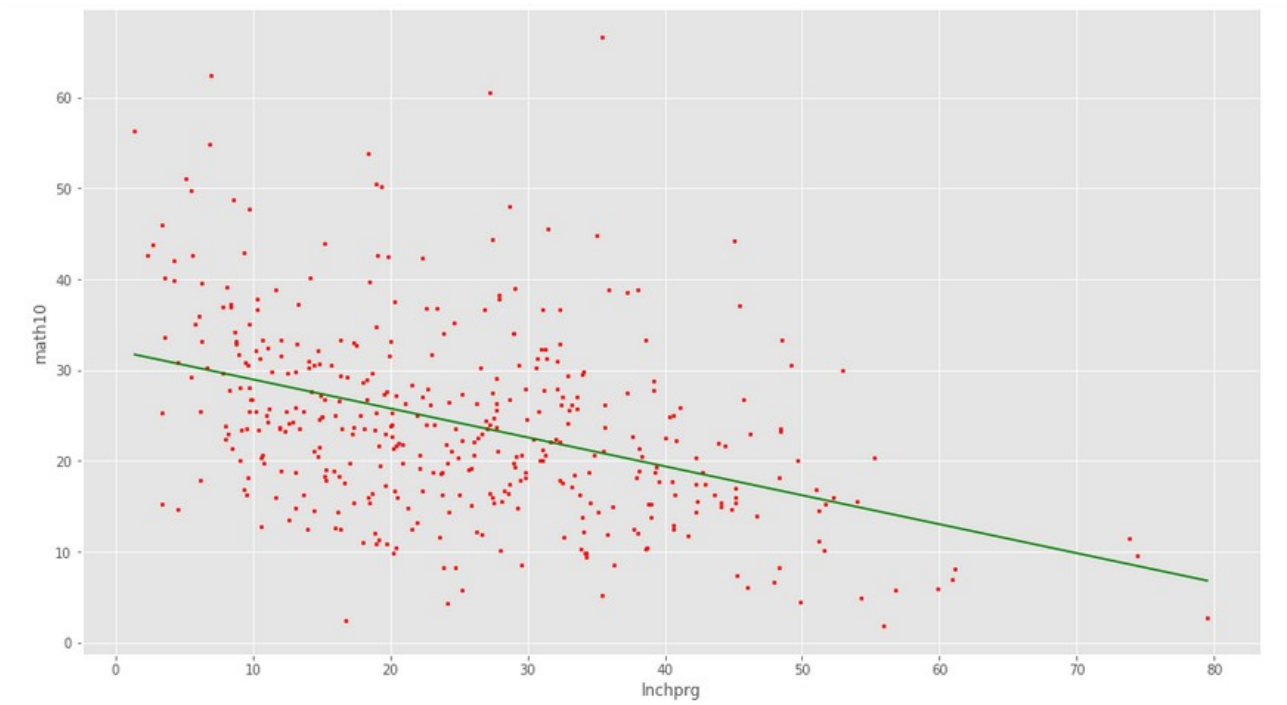
so we have to solve $\begin{cases} (\sum_i 2)\beta_0+(\sum_i 2\,lnchprg_i)\beta_1+(\sum_i(-2\,math10_i))=0 \\ (\sum_i 2lnchprg_i)\beta_0+(\sum_i 2\,lnchprg_i^2)\beta_1+(\sum_i(-2\,math10_i\,lnchprg_i))=0 \end{cases}$

$\Leftrightarrow \begin{cases} (\sum_i 2)\beta_0+(\sum_i 2\,lnchprg_i)\beta_1=\sum_i(2\,math10_i) \\ (\sum_i 2\,lnchprg_i)\beta_0+(\sum_i 2\,lnchprg_i^2)\beta_1=\sum_i(2\,math10_i\,lnchprg_i) \end{cases}$

0845209
Siméon Gaumart

so we have to solve $\begin{pmatrix} \sum_i 2 & \sum_i 2\,lnchprg_i \\ \sum_i 2\,lnchprg_i & \sum_i 2\,lnchprg\,i^2 \end{pmatrix} \times \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = \begin{pmatrix} \sum_i 2\,math10_i \\ \sum_i 2\,math10_i\,lnchprg_i \end{pmatrix}$

we found, with python, that math10 = 32.14271164450065 + -0.31886428597057154 * lnchprg



in green : the curve math10 = 32.14271164450065 + -0.31886428597057154 * lnchprg
in red : the datas (math10, lnchprg) used to calculate the regression equation

b) and c)
We have the equation math10 = 32.14271164450065 + -0.31886428597057154 * lnchprg

So for lnchprg_a we have math10_a = 32.14271164450065 + -0.31886428597057154 * lnchprg_a.

Now if we have lnchprg_b = lnchprg_a + 10%*lnchprg_a,
We have
math10_b = 32.14271164450065 + -0.31886428597057154 * lnchprg_b
= 32.14271164450065 + -0.31886428597057154 * (lnchprg_a + 10%*lnchprg_a)
= math10_a + -0.31886428597057154 * 10%lnchprg_a

So the math10_b – math10_a = -0.31886428597057154 * 10%*lnchprg_a
= -0.031886428597057154 * lnchprg_a

So an increment of 10% in the number of student eligible for the lunch program in an university decrease the percentage of student passing the math exam of 0.031886428597057154 *lnchprg (so it depends on the percentage of students eligible for the lunch program before the increment, indeed, the function should be logarithmic to have a static variation of the percentage of math10 each time we increment lnchprg of 10%).

0845209
Siméon Gaumart

d)

we have to estimate $\hat{\beta}_0$, $\hat{\beta}_1$ and $\hat{\beta}_2$ in the equation

$\hat{math}10 = \hat{\beta}_0 + \hat{\beta}_1 * \log(expend) + \hat{\beta}_2 * lnchprg.$

We introduce $X = \begin{pmatrix} 1 & \log(expend)_1 & lnchprg_1 \\ 1 & \log(expend)_2 & lnchprg_2 \\ \vdots & \vdots & \vdots \\ 1 & \log(expend)_n & lnchprg_n \end{pmatrix}$ $Y = \begin{pmatrix} math10_1 \\ math10_2 \\ \vdots \\ math10_n \end{pmatrix}$ $\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix}$

So here, the loss function would be

$S = (Y - \beta^T X)^T (Y - \beta^T X)$

$\Leftrightarrow S = Y^T Y + X^T \beta \beta^T X - 2 X^T \beta Y$

So $\frac{\partial S}{\partial \beta} = 2 X^T X \beta - 2 X^T Y$

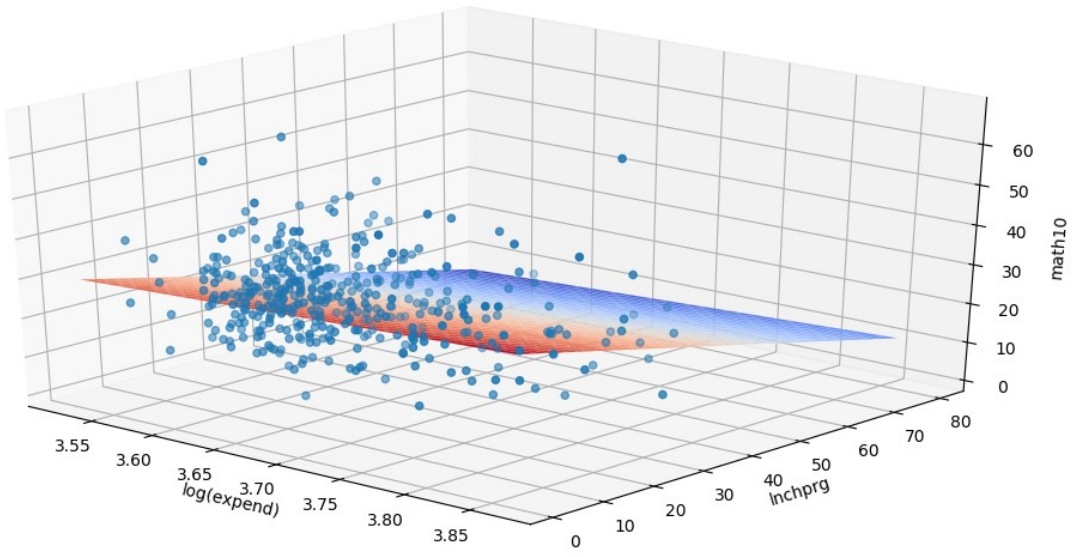
And we have to solve $\frac{\partial S}{\partial \beta} = 0$

$\Leftrightarrow 2 X^T X \beta - 2 X^T Y = 0$

$\Leftrightarrow \beta = (X^T X)^{-1} X^T Y$

We found, with python, that

$math10 = -20.360816474189132 + 14.344410366129082 * \log(expend) + -0.30458530220642444 * lnchprg$



the plan is our linear regression. Plan of equation
$math10 = -20.360816474189132 + 14.344410366129082 * \log(expend) + -0.30458530220642444 * lnchprg$
the points in blue are the data (x=log(expend),y=lnchprg,z=math10)

It means that the percentage of tenth graders at a high school receiving a passing score will increase with an augmentation of the expends but decrease with an augmentation of the percentage of students eligible for the lunch program.

and R-squared is $R^2 = 0.17992710085133723 \approx 18\%$

We can see that R^2 is low. Maybe we should use an other model, and not a linear regression.

Problem 3 :

a)
What are the relationships between data, informations, knowledge and wisdom ?

Firstly, we start from the data. A data is one or many words, characters, and/or numbers associated with labels, which define something.
Example of data : 1012, rains, Toulouse, 15
From the data, we can deduce informations, which are the meaning of the data. A literal way to read data.
Example of information : It rains at Toulouse, the atmospheric pressure is low and the temperature is moderate.
A knowledge get the parts of the informations which are interesting, depending on the context knowledge's owner.
Example of knowledge : I'm leaving my house and it rains outside.
Finally, from the knowledge, we can deduce what to do, the action to take. This is the wisdom.
Example of wisdom : I better get an umbrella !

However, from the wisdom, we can't deduce the knowledge, from the knowledge we can't deduce all the informations, and from the informations we can't deduce the data.
Example : From getting an umbrella, we can't deduce that it rains (it could be to sunny too, or the person could need it for a cosplay for example). From the knowledge that the person leave her house and it rains, we can't deduce her location (Toulouse) and the rest of the informations (pressure and temperature). From the informations that the pressure is low and the temperature moderate, we can't deduce the real values of this parameters int the data.
From the data to the wisdom, the quantity of information is decreasing, but we get closer and closer to a solution.

b)
What is the difference between supervised learning and unsupervised learning?

In supervised learning, a machine, before being able to predict, needs to be train with labeled datas to define and tweak the parameters of the model chosen. When the training is sufficient, the machine is able to predict te label of unlabeled datas.

0845209
Siméon Gaumart

Example of supervised learning : We can imagine an algorithm (which already exists) on a music application which could get the musics liked or unliked by an user and train its model with this datas to determine, after, which musics this user could like.

In unsupervised learning, a machine doesn't use labeled data to train. It analyzes the structure of the datas, extract the features to define, at the end, a model which best fits with this data structure. And uses it to predict.
Example of unsupervised learning : We want, for example, classify animal photos on a website, given by users. But we don't know what is the animal on each photo. But we know the number of different animals. We must use unsupervised learning here, giving all the photos to the algorithm and let it cluster the photos in the good groups.

Problem 4 :

a)
This paper, from Alex Krizhevsky, Ilya Sutskever and Geoffrey E. Hinton, speaks about their work on a convolutional neural network, used to classify a big amount of images. In this paper, they speak about what they did to make their convolutional neural network more efficient and how reached good results.
This paper gives a technical contribution to the machine learning world with providing new solutions of optimization in the architecture of the convolutional neural network, with a new model for the neural network, a way of using the graphic processors to optimize the neural network training, etc. It provides too some ways to reduce overfitting with, for example, get more data with transforming the originals datas (transform an image with a python code), etc.
They explain too how they set the data, how they train their neural network to get the bests results, and they show this results to prove the efficiency of their methods.

b)
This work on a convolutional neural network, follows the machine learning procedure :
Defining problem : The problem is to classify a big amount of images. It's a common problem in machine learning. That's why they didn't give so much details on the problem definition.
Gathering data : To gather data, they use an already existing dataset, ImageNet. It's a dataset of over 15 millions images labeled by humans.
Data preparation : To use the datas they gathered, they had to change the resolution of each images to 250x250 with resizing the image and make it square. Furthermore, to avoid overfitting, they used this datas to create other images and have a larger dataset.
Model development : In the architecture presentation, they speak about the model they used, ReLU. It is not the conventional model for convolutional neural network, but it converges to a lower error rate faster than the standard one. It's more efficient than the standard.
Training : To train the neural network, they use two graphic processors in parallel to train a bigger network than usually. They use too a stochastic gradient descent for the training.
Evaluation : They trained and evaluated their neural network with the ImageNet dataset (one part of the dataset to train and the other part for evaluation) and reached the best results ever on this dataset. They evaluated their neural network with calculating the percentage of error with giving one answer or five answers for each image.
Paraeter tuning : This convolutional neural network have 60 millions parameters to tune. They're tuned during the training at each iteration, following a weight function.