Machine Learning Engineer
Take-Home Challenge

WorldQuant Predictive Research Technology Team

Contents

1 Instructions

In the following section you will find two problems. Please read the problems carefully and provide your best answers.

Part of the purpose of this challenge is to get to know you as an engineer, how you approach and solve problems, and how you present your solutions. For that reason, if not explicitly stated in the problem specification, you may submit your solutions in any format you like (including, where appropriate, handwritten as long as it is legible). Please use your best judgment regarding how you document your solutions and how much detail you provide.

2 Problems

2.1 Short answer question

This problem concerns the following situation:

> At a recent meeting of the International Conference of Useless Statistics held by the Intergovernmental Panel of Practitioners of Impractical Practices (IPPIP) it was determined to be a top priority to know the average height of all currently living humans, which they denote as H_{living}. They were in unanimous agreement that a scientific study should be carried out to determine H_{living} once and for all.

Additionally, it was decided that the best way to figure out how to determine this number would be to solicit proposals from the public at large on how best to determine it. The IPPIP announced their call for submissions of proposals in all major news venues, stating that proposals should address potential challenges and obstacles in obtaining a value for H_{living}, and that proposal assessments shall in no way be concerned with time or cost of the proposed study (as these are essentially *limitless* for the IPPIP), should it actually be implemented. Their call for submissions repeatedly indicated the dire importance of *accuracy* in measuring H_{living}.

IPPIP has a *virtually unlimited* budget for such important questions, and has therefore declared that the member of the public selected with the winning proposal shall receive a generous cash award of 100 million dollars, USD, payable immediately.

Write a proposal for the IPPIP not exceeding min(1 page, 2 paragraphs).

2.2 Coding exercise

For this question, we will consider the simple linear least squares problem. In this type of learning problem, training data is given in the form $\{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{m}$, with $\mathbf{x}^{(i)} \in \mathbb{R}^n$ the feature vectors and $y^{(i)} \in \mathbb{R}$, the corresponding targets. The linear least squares problem hypothesizes a few things, but relevant to this question, are the following[1]:

- $m > n$, namely, there are more examples than there are features.
- A linear relationship between features and targets might reasonably exist, namely, that the *approximate* matrix equation

  $$D\theta \approx \mathbf{y}$$

  should hold, despite the exact equation $D\theta = \mathbf{y}$ failing to obtain. Here, $\theta \in \mathbb{R}^{n+1}$ is a vector of unknown constants, $\mathbf{y} = (y^{(1)}, ..., y^{(n)})^T$ is the vector of targets and the so-called "design matrix", D, has m rows and $n + 1$ columns, taking the form

  $$D = \begin{bmatrix} (1, \mathbf{x}_1^{(1)}, \mathbf{x}_2^{(1)}, ..., \mathbf{x}_n^{(1)}) \\ (1, \mathbf{x}_1^{(2)}, \mathbf{x}_2^{(2)}, ..., \mathbf{x}_n^{(2)}) \\ \vdots \\ (1, \mathbf{x}_1^{(m)}, \mathbf{x}_2^{(m)}, ..., \mathbf{x}_n^{(m)}) \end{bmatrix}$$

  In the above, the notation $\mathbf{x}_j^{(i)}$ indicates the j'th entry of the i'th example vector.

[1] For this question, we are ignoring whether something like ordinary least squares is appropriate for the problem at hand. We are only looking about the assumptions relevant for programming a suitable predictor

In looking for an approximate solution to $D\theta = \mathbf{y}$, linear least squares seeks to minimize the quantity $||D\theta - \mathbf{y}||^2$. In other words, it attempts to solve

$$\hat{\theta} = \arg\min_{\theta} ||D\theta - \mathbf{y}||^2$$

Here, the notation $||\cdot||$, denotes the standard Euclidean distance function in $\mathbb{R}^m$, namely $||\mathbf{a}|| = \sqrt{a_1^2 + \cdots + a_m^2}$. Because of this, we view the function

$$J(\theta) = \frac{1}{2}||D\theta - \mathbf{y}||^2$$

as the *loss function* that we wish to minimize.

Your task consists of the following:

1. Come up with a gradient descent algorithm to use for the problem just described, with loss function given by $J(\theta)$.
2. In the attached Python file linear_model.py there is an abstract class given for a linear model. Your job is to create a subclass, LinearLeastSquares, from this base class, which implements a linear model object having both a fit and a predict method. The fit method should be based on solving the linear least squares problem described above *using the gradient descent method* found in the previous question. The input for both of these methods should be an array, X, containing the examples as rows. Importantly, this input is **not** the design matrix mentioned above. This is not the "normal" way to solve linear least squares, the point is to use gradient descent.

Your solution should take the form of a filled out linear_model.py file together with any additional .py files required to run your code, if any. Please do not submit any .ipynb files. As well, your solution should not make use of any imported libraries other than the two already given in the header of linear_model.py.