FIS - Statistical Machine Learning
Assignment 1

Please send me

- the **original script** detailing your computations.
  - The script must be **documented**, i.e. the code corresponding to each answer must be delimited and your loops/variables briefly explained.
  - The script must be **executable**: by just running your script, all results should appear **automatically**.
  - Do not use external functions, everything must be coded **by yourself** using elementary linear algebra functions.
- A document (.doc, .pdf) which will contain your answer and your analysis. Do not put your source code in that document. Illustrations, graphs, *etc.*are welcome.

This homework is due **June 12th (Sun.) 23:59 PM**

Send your homework at mcuturi@i.kyoto-u.ac.jp

Least-square and Locally-weighted least-square regression

- Download the dataset available on `http://data.princeton.edu/wws509/datasets/#births` (file `phbirths.raw`), read its description and import it into your software of choice.
- Divide the dataset into 2 folds of equal size. The first fold will be called the `train` fold, the second will be called the `test` fold.
- Estimate a vector β and a constant b such that

$$y \approx \beta^T \mathbf{x} + b.$$

  where y is the Birth weight in grams, using the data available in the `train` fold and least-square regression.
- Compare the average error these coefficients yield on both the `train` fold and the `test` fold, give a brief interpretation for these coefficients.

Given a `train` database of points $\{(\mathbf{x}_i, y)\}_{i=1,\cdots,n}$ where $\mathbf{x}_i \in \mathbb{R}^d$ and $y \in \mathbb{R}$, least-square regression finds the minimizer of

$$(\beta_\star, b_\star) = \underset{\beta,b}{\operatorname{argmin}} \sum_{i=1}^{n} \left\| y_i - \begin{bmatrix} b & \beta^T \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix} \right\|^2$$

to predict, given a new point $\mathbf{x}_{\text{new}}$, its corresponding predicted variable as $\begin{bmatrix} b_\star & \beta_\star^T \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{x}_{\text{new}} \end{bmatrix}$. A different technique, called locally-weighted linear locally regression, tries to exploit the similarity of the point we are interested in, $\mathbf{x}_{\text{new}}$, with respect to other points in the database,

$$w_i \stackrel{\text{def}}{=} \text{similarity}(\mathbf{x}_{\text{new}}, \mathbf{x}_i), \quad i = 1, \cdots, n$$

by defining instead

$$(\beta_\sharp, b_\sharp) = \underset{\beta, b}{\text{argmin}} \sum_{i=1}^{n} w_i \left\| y_i - \begin{bmatrix} b & \beta \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix} \right\|^2,$$

and using $(\beta_\sharp, b_\sharp)$ to predict the corresponding y variable of $\mathbf{x}_{\text{new}}$ as $\begin{bmatrix} b_\sharp & \beta_\sharp^T \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{x}_{\text{new}} \end{bmatrix}$

- Compute the average error of locally weighted regression on the `test` fold, assuming

  $$\text{similarity}(\mathbf{x}, \mathbf{x}') = e^{-(\mathbf{x}-\mathbf{x}')^T \Sigma^{-1} (\mathbf{x}-\mathbf{x}')/2},$$

  where Σ is the empirical variance matrix of your `train` fold, namely

  $$\Sigma = \frac{1}{n_{\text{train}} - 1} \sum_{i=1}^{n_{\text{train}}} \left(\mathbf{x}_i - \frac{1}{n_{\text{train}}} \sum_{j=1}^{n_{\text{train}}} \mathbf{x}_j \right) \left(\mathbf{x}_i - \frac{1}{n_{\text{train}}} \sum_{j=1}^{n_{\text{train}}} \mathbf{x}_j \right)^T.$$

  In order to do so, you will have to compute a different $(\beta_\sharp, b_\sharp)$ for **each** element of the test fold. Explain how you can compute $(\beta_\sharp, b_\sharp)$.

- What are the advantages/disadvantages of locally-weighted regression compared to standard regression?