Homework 1: Linear Regression

Writeup due 23:59 on Friday 6 February 2015

You will do this assignment individually and submit your answers as a PDF via the Canvas course website. There is a mathematical component and a programming component to this homework.

1. Non-Uniformly Weighted Data [7pts]

Consider a data set in which each data point t_n is associated with a weighting factor $r_n > 0$, so that the sum-of-squares error function becomes

$$E_D(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N} r_n\{t_n - \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_n)\}^2.$$

Find an expression for the solution $\boldsymbol{w}^*$ that minimizes this error function.

Solution 1: Take the gradient and set to zero:

$$\frac{\partial}{\partial \boldsymbol{w}} E_D(\boldsymbol{w}) = -\sum_{n=1}^{N} r_n\{t_n - \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_n)\}\boldsymbol{\phi}(\boldsymbol{x}_n) = 0.$$

Solve for $\boldsymbol{w}$:

$$\sum_{n=1}^{N} r_n t_n \boldsymbol{\phi}(\boldsymbol{x}_n) = \left(\sum_{n=1}^{N} r_n \boldsymbol{\phi}(\boldsymbol{x}_n)\boldsymbol{\phi}(\boldsymbol{x}_n)^\mathsf{T}\right)\boldsymbol{w}$$

$$\boldsymbol{w} = \left(\sum_{n=1}^{N} r_n \boldsymbol{\phi}(\boldsymbol{x}_n)\boldsymbol{\phi}(\boldsymbol{x}_n)^\mathsf{T}\right)^{-1}\left(\sum_{n=1}^{N} r_n t_n \boldsymbol{\phi}(\boldsymbol{x}_n)\right)$$

Solution 2 (Matrix form): We rewrite the error function in terms of matrix products:

$$\begin{aligned} E_D(\boldsymbol{w}) &= \frac{1}{2}\sum_{n=1}^{N} r_n\{t_n - \boldsymbol{w}^\mathsf{T}\boldsymbol{\phi}(\boldsymbol{x}_n)\}^2 \\ &= \frac{1}{2}(\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{t})^\mathsf{T}\boldsymbol{R}(\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{t}) \\ &= \frac{1}{2}(\boldsymbol{w}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{w}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{t} - \boldsymbol{t}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{t}^\mathsf{T}\boldsymbol{R}\boldsymbol{t}) \\ &= \frac{1}{2}(\boldsymbol{w}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi}\boldsymbol{w} - 2\boldsymbol{t}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{t}^\mathsf{T}\boldsymbol{R}\boldsymbol{t}) \end{aligned}$$

Where we have defined $\boldsymbol{R} = \mathrm{diag}(r_1, \cdots, r_N)$.
Taking the gradient of the error function:

$$\begin{aligned} \nabla E_D(\boldsymbol{w}) &= \boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi}\boldsymbol{w} - \boldsymbol{t}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi} \\ \boldsymbol{w} &= (\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi})^{-1}\boldsymbol{t}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi} \\ &= (\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{R}\boldsymbol{t} \end{aligned}$$

2. Priors and Regularization [7pts]

Consider the Bayesian linear regression model given in Bishop 3.3.1. The prior is

$$p(\boldsymbol{w} \mid \alpha) = \mathcal{N}(\boldsymbol{w} \mid \mathbf{0}, \alpha^{-1}\boldsymbol{I}),$$

where α is the precision parameter that controls the variance of the Gaussian prior. The likelihood can be written as

$$p(\boldsymbol{t} \mid \boldsymbol{w}) = \prod_{n=1}^{N} \mathcal{N}(t_n \mid \boldsymbol{w}^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{x}_n), \beta^{-1}),$$

Using the fact that the posterior is the product of the prior and the likelihood (up to a normalization constant), show that maximizing the log posterior (i.e., $\ln p(\boldsymbol{w} \mid \boldsymbol{t}) = \ln p(\boldsymbol{w}|\alpha) + \ln p(\boldsymbol{t} \mid \boldsymbol{w})$) is equivalent to minimizing the regularized error term given by $E_D(\boldsymbol{w}) + \lambda E_W(\boldsymbol{w})$ with

$$E_D(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}(t_n - \boldsymbol{w}^{\mathsf{T}}\boldsymbol{\phi}(\boldsymbol{x}_n))^2$$
$$E_W(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^{\mathsf{T}}\boldsymbol{w}$$

Do this by writing $\ln p(\boldsymbol{w} \mid \boldsymbol{t})$ as a function of $E_D(\boldsymbol{w})$ and $E_W(\boldsymbol{w})$, dropping constant terms if necessary. Conclude that maximizing this posterior is equivalent to minimizing the regularized error term given by $E_D(\boldsymbol{w}) + \lambda E_W(\boldsymbol{w})$. (Hint: take $\lambda = \alpha/\beta$)

Solution: $p(\boldsymbol{w} \mid \alpha)$ is a multivariate normal distribution. Suppose the dimension of $\boldsymbol{w}$ is $D \times 1$. Plug the mean $\mathbf{0}$ and covariance matrix $\alpha^{-1}\boldsymbol{I}$ into the PDF of multivariate normal distribution:

$$p(\boldsymbol{w} \mid \alpha) = \frac{1}{\sqrt{(2\pi)^D det(\alpha^{-1}\boldsymbol{I})}} exp(-\frac{1}{2}\boldsymbol{w}^T(\alpha^{-1}\boldsymbol{I})^{-1}\boldsymbol{w})$$
$$\ln p(\boldsymbol{w} \mid \alpha) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}\alpha + constant = -\alpha E_W(\boldsymbol{w}) + constant$$

Similarly,

$$p(\boldsymbol{t} \mid \boldsymbol{w}) = \frac{1}{\sqrt{2\beta^{-1}\pi}} \prod_{n=1}^{N} exp(-\frac{(t_n - \boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x}_n))^2}{2\beta^{-1}})$$
$$\ln p(\boldsymbol{t} \mid \boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}(t_n - \boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x}_n))^2\beta + constant = -\beta E_D(\boldsymbol{w}) + constant$$

Therefore, maximizing $\ln p(\boldsymbol{w}|\alpha) + \ln p(\boldsymbol{t} \mid \boldsymbol{w})$ is equivalent to maximizing $-\beta E_D(\boldsymbol{w}) - \alpha E_W(\boldsymbol{w})$. Hence it is equal to minimizing $E_D(\boldsymbol{w}) + \lambda E_W(\boldsymbol{w})$, where $\lambda = \alpha/\beta$. (Note that $\beta > 0$ because it is a variance.)

3. Modeling Motorcycle Helmet Forces [10pts]

The objective of this problem is to learn about linear regression with basis functions by modeling the g-forces associated with motorcycle helmet impacts. Download the file `motorcycle.csv` from the course website. It has two columns. The first one is the number of milliseconds since impact and the second is the g-force on the head. The data file looks like this:

```
"time since impact (ms)","g force"
2.4,0
2.6,-1.3
3.2,-2.7
3.6,0
4,-2.7
6.2,-2.7
6.6,-2.7
6.8,-1.3
...
```

and you can see a plot of the data in Figure 1.

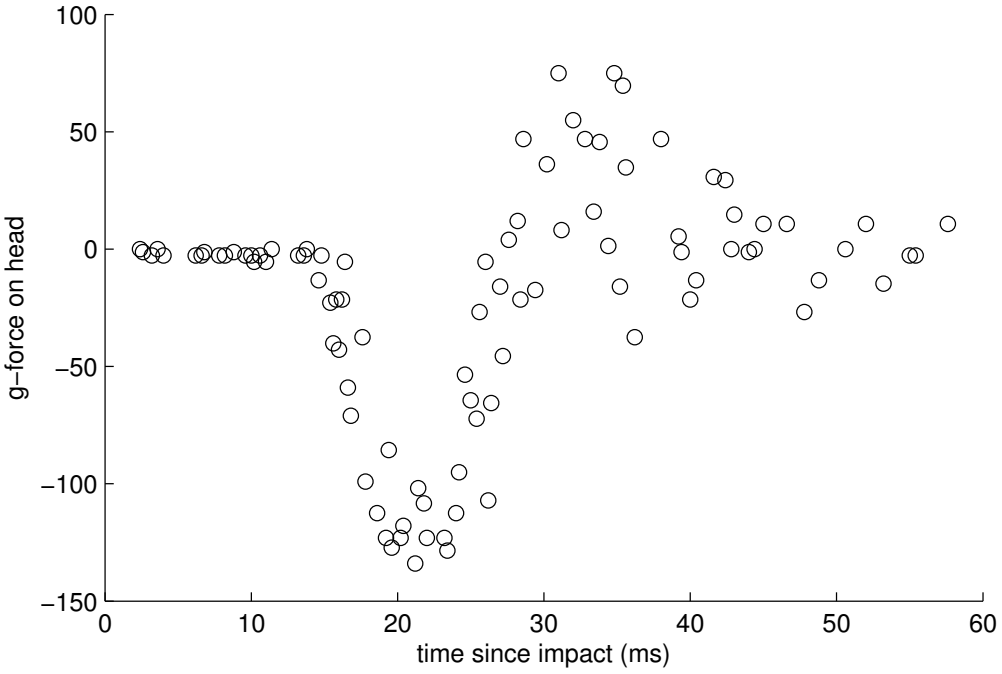


Figure 1: Motorcycle crash helmet data. The horizontal axis is time since impact and the vertical axis is force on the head.

Implement basis function regression with ordinary least squares.[1] Some sample Python code is provided in `linreg.py`. Plot the data and regression lines for the simple linear case, and for each of the following sets of basis functions:

(a) $\phi_j(x) = \exp\{-((x-10j)/5)^2\}$ for $j = 0, \ldots, 6$

(b) $\phi_j(x) = \exp\{-((x-10j)/10)^2\}$ for $j = 0, \ldots, 6$

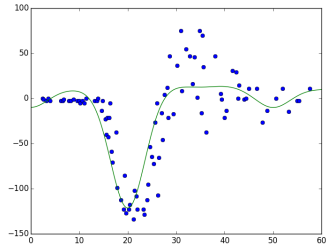
(c) $\phi_j(x) = \exp\{-((x-10j)/25)^2\}$ for $j = 0, \ldots, 6$

(d) $\phi_j(x) = x^j$ for $j = 0, \ldots, 10$

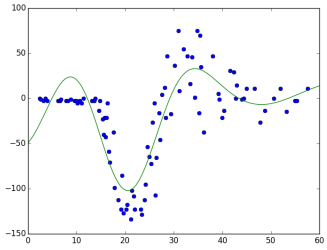
(e) $\phi_j(x) = \sin\{x/j\}$ for $j = 1, \ldots, 20$

In addition to the plots, provide one or two sentences for each, explaining whether you think it is fitting well, overfitting or underfitting. If it does not fit well, provide a sentence explaining why.

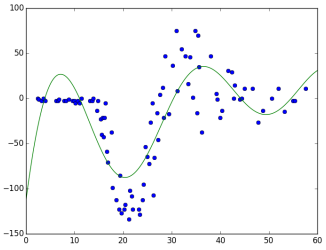
Solution: See the posted solution code.



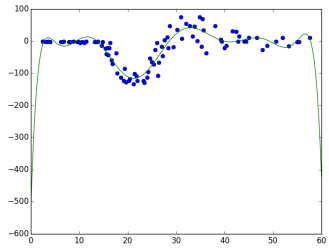
(a)



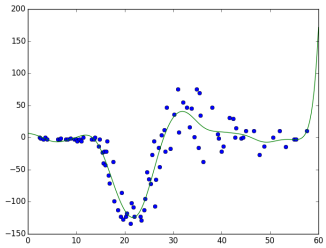
(b)



(c)



(d)



(e)

(a) fits relatively well. (b) and (c) exhibit the problem of underfitting: the flat tail on the left is not properly fit in these two plots. If we plot the basis functions for (b) and (c), we can see that they are wider than (a), and cannot capture the complexity in the data

[1] Note that the data clearly don't have fixed variance! There is obviously less variance on the left of the plot. Modeling such *heteroscedastic* data is beyond the scope of the course.

pattern as well. (d) and (e) are overfitting: they fit the given data well, but also have captured the noise. The steep rising / falling tails are not likely patterns of the original data. This is because we fit the data to high-degree basis functions (10th and 20th degree respectively).

Calibration [1pt]

Approximately how long did this homework take you to complete?

Changelog

- **v1.0** – 28 January 2015 at 13:00
- **v1.1** – 28 January 2015 at 22:30. Removed extra λ in $E_W(\boldsymbol{w})$, minor formatting edits