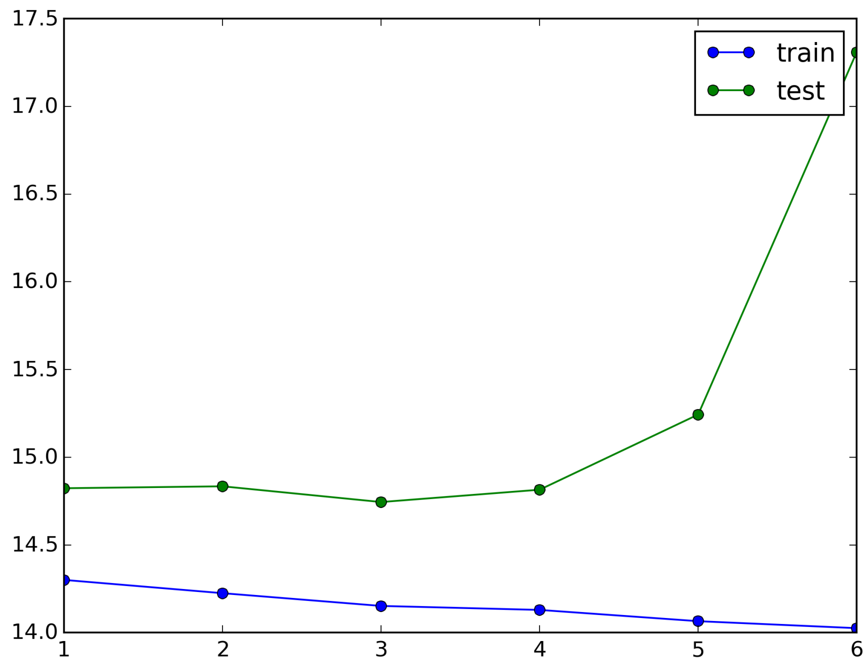
EECS 545 – Machine Learning - Homework #2

David Ke Hong Due: 11:00pm 02/08/2016

1) **Linear Regression (20 pts).**

(a)

Figure 1: 1.png



(b)

(c)

2) **Open Kaggle challenge (15 pts).** See the online ranking

3) **Weighted Linear Regression (15 pts).**

Figure 2: 2.png

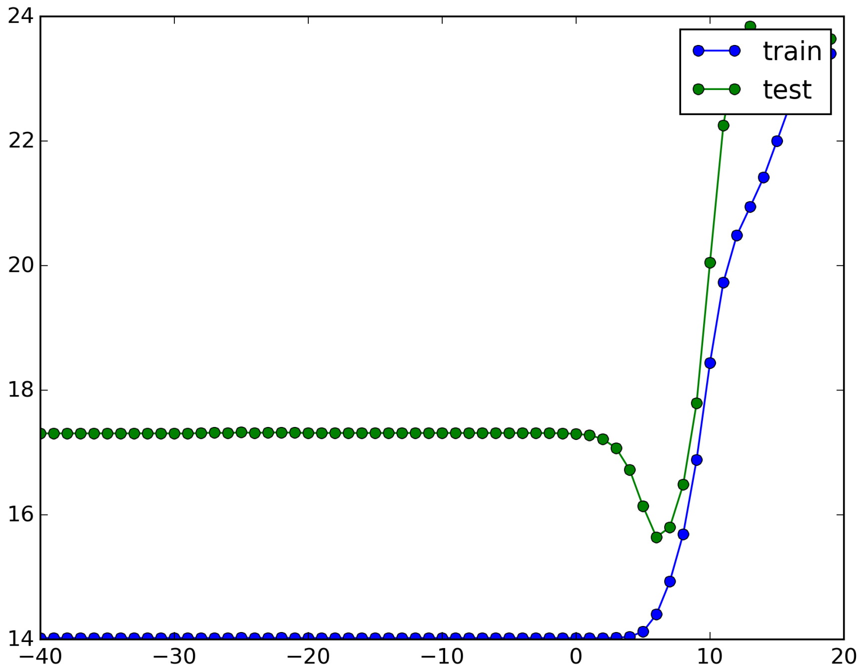
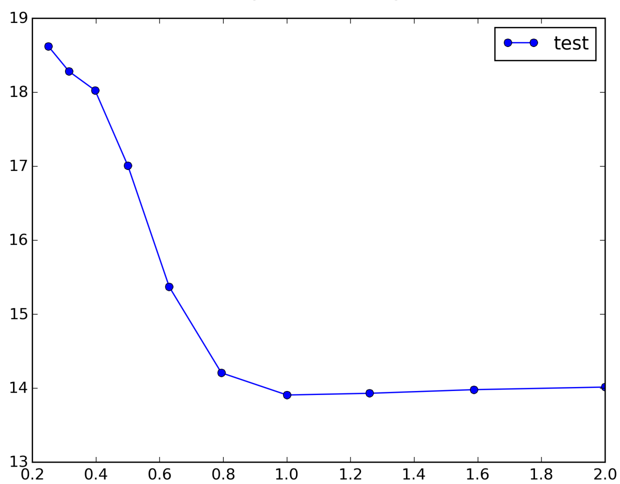


Figure 3: 3.png



(a) Define

$$X_{N\times M} = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \end{bmatrix}^\top$$
$$R_{N\times N} = \mathrm{diag}(\frac{r_1}{2}, \frac{r_2}{2}, \cdots, \frac{r_N}{2})$$
$$t_{N\times 1} = \begin{bmatrix} t_1 & t_2 & \cdots & t_N \end{bmatrix}^\top$$

Now

$$Xw - t = \begin{bmatrix} x_1^\top w - t_1 \\ x_2^\top w - t_2 \\ \vdots \\ x_N^\top w - t_N \end{bmatrix} = \begin{bmatrix} w^\top x_1 - t_1 \\ w^\top x_2 - t_2 \\ \cdots \\ w^\top x_N - t_N \end{bmatrix} \qquad \text{(note: } x_i^\top w = w^\top x_i\text{)}$$

Hence

$$\begin{aligned} & (Xw - t)^\top R(Xw - t) \\ = \; & \begin{bmatrix} w^\top x_1 - t_1 & w^\top x_2 - t_2 & \cdots & w^\top x_N - t_N \end{bmatrix} \begin{bmatrix} \frac{r_{11}}{2} & & & \\ & \frac{r_{22}}{2} & & \\ & & \ddots & \\ & & & \frac{r_{NN}}{2} \end{bmatrix} \begin{bmatrix} w^\top x_1 - t_1 \\ w^\top x_2 - t_2 \\ \cdots \\ w^\top x_N - t_N \end{bmatrix} \\ = \; & \frac{1}{2}\sum_{i=1}^{N} r_i (w^\top x_i - t_i)^2 = E_D(w) \end{aligned}$$

(b)

$$\begin{aligned} \nabla_w E_D(w) &= \nabla_w \left\{ (Xw - t)^\top R(Xw - t) \right\} \\ &= \nabla_w \left\{ w^\top X^\top RXw - 2t^\top RXw + t^\top Rt \right\} \\ &= 2X^\top RXw - 2(t^\top RX)^\top && \text{(Gradient of quadratic form and linear function)} \\ &= 2X^\top RXw - 2X^\top Rt && (R \text{ is diagonal thus symmetric}) \end{aligned}$$

Setting $\nabla_w E_D(w) = 0$, we get $w^* = (X^\top RX)^{-1} X^\top Rt$.

(c)

$$\begin{aligned} \arg\max_w \prod_{i=1}^{N} p(t_i|\mathbf{x_i}, \mathbf{w}) &= \arg\max_w \sum_{i=1}^{N} \log(p(t_i|\mathbf{x_i}, \mathbf{w})) \\ &= \arg\max_w \sum_{i=1}^{N} \log(\frac{1}{\sqrt{2\pi}\sigma_i}) - \frac{(t_i - \mathbf{w}^T\mathbf{x_i})^2}{2(\sigma_i)^2} \\ &= \arg\min_w \frac{1}{2}\sum_{i=1}^{N} \frac{1}{\sigma_i^2}(\mathbf{w}^T\mathbf{x_i} - t_i)^2 \\ &= \arg\min_w \frac{1}{2}\sum_{i=1}^{N} r_i(\mathbf{w}^T\mathbf{x_i} - t_i)^2 \end{aligned}$$

Therefore, maximizing the log-likelihood reduces to solving a weighted linear regression problem where $r_i = 1/\sigma_i^2$.

4) **Naive Bayes Classifier (35 pts).**

(a)

(i) The pre-processing step cannot be applied to nominal (or categorical) feature variables. In spambase, features are word frequencies, character frequencies, length of capital letter in a row, and the number of capital letters, none of which are nominal.

(ii) The test error of Naive Bayes classifier is around 10.8%, and the baseline algorithm predicting the majority class has test error around 38.6%.

(b) See the online ranking

5) **Softmax Regression (15 pts).**

(a)

$$
\begin{aligned}
E(\mathbf{w}) &= -\log p(\mathbf{t}|\mathbf{w}) \\
&= -\sum_{n=1}^{N}\sum_{k=0}^{K-1} \mathbf{1}(t_n = k)\log p(C_k|\phi(\mathbf{x}_n)) \\
&= -\sum_{n=1}^{N}\sum_{k=0}^{K-1} \mathbf{1}(t_n = k)(\mathbf{w_k^T}\phi(\mathbf{x}_n) - \log\sum_{j=0}^{K-1}\exp(\mathbf{w}_j^T\phi(\mathbf{x}_n)) \\
&= -\sum_{n=1}^{N}(\mathbf{w}_{t_n}^T\phi(\mathbf{x}_n) - \log\sum_{j=0}^{K-1}\exp(\mathbf{w}_j^T\phi(\mathbf{x}_n)) \\
\nabla_{\mathbf{w}_j}E(\mathbf{w}) &= -\nabla_{\mathbf{w}_j}\sum_{n=1}^{N}(\mathbf{w}_{t_n}^T\phi(\mathbf{x}_n) - \log\sum_{j=0}^{K-1}\exp(\mathbf{w}_j^T\phi(\mathbf{x}_n)) \\
&= -\sum_{n=1}^{N}(\mathbf{1}(t_n = j)\phi(\mathbf{x}_n) - \frac{\exp(\mathbf{w}_j^T\phi(\mathbf{x}_n))\phi(\mathbf{x}_n)}{\sum_{j=0}^{K-1}\exp(\mathbf{w}_j^T\phi(\mathbf{x}_n))}) \\
&= -\sum_{n=1}^{N}(\mathbf{1}(t_n = j) - p(C_j|\phi(\mathbf{x}_n))\phi(\mathbf{x}_n)
\end{aligned}
$$

(b)

$$
\nabla_{\mathbf{w}_j}E^{\lambda}(\mathbf{w}) = -\sum_{n=1}^{N}(\mathbf{1}(t_n = j) - p(C_j|\phi(\mathbf{x}_n))\phi(\mathbf{x}_n) + \lambda\mathbf{w}_j
$$