Multiplicative updates for the LASSO

Mørup, Morten; Clemmensen, Line Katrine Harder

Published in:
2007 IEEE International Workshop on MACHINE LEARNING FOR SIGNAL PROCESSING

Link to article, DOI:
10.1109/MLSP.2007.4414278

Publication date:
2007

Document Version
Early version, also known as pre-print

MULTIPLICATIVE UPDATES FOR THE LASSO

Morten Mørup and Line Harder Clemmensen

Technical University of Denmark
Informatics and Mathematical Modelling
Richard Petersens Plads, Building 321
DK-2800 Kgs. Lyngby, Denmark
email: {mm,lhc}@imm.dtu.dk

ABSTRACT

Multiplicative updates have proven useful for non-negativity constrained optimization. Presently, we demonstrate how multiplicative updates also can be used for unconstrained optimization. This is for instance useful when estimating the least absolute shrinkage and selection operator (LASSO) i.e. least squares minimization with L_1-norm regularization, since the multiplicative updates (MU) can efficiently exploit the structure of the problem traditionally solved using quadratic programming (QP). We derive two algorithms based on MU for the LASSO and compare the performance to Matlabs standard QP solver as well as the basis pursuit denoising algorithm (BP) which can be obtained from www.sparselab.stanford.edu. The algorithms were tested on three benchmark bio-informatic datasets: A small scale data set where the number of observations is larger than the number of variables estimated ($M < J$) and two large scale microarray data sets ($M \gg J$). For small scale data the two MU algorithms, QP and BP give identical results while the time used is more or less of the same order. However, for large scale problems QP is unstable and slow. both algorithms based on MU on the other hand are stable and faster but not as efficient as the BP algorithm and converge slowly for small regularizations. The benefit of the present MU algorithms is that they are easy to implement, they bridge multiplicative updates to unconstrained optimization and the updates derived monotonically decrease the cost-function thus does not need any objective function evaluation. Finally, both MU are potentially useful for a wide range of other models such as the elastic net or the fused LASSO. The Matlab implementations of the LASSO based on MU can be downloaded from [1].

1. INTRODUCTION

Multiplicative updates were introduced to solve the non-negative matrix factorization (NMF) problem, i.e. factor analysis with non-negativity constraints imposed on all variables [2, 3]. This has recently been extended to semi-NMF, i.e. where the parameters under consideration are non-negative while the data in itself is unconstrained [4, 17]. We will presently advance the multiplicative updates to unconstrained optimization, i.e. problems where the parameters can both take positive and negative values. We demonstrate, that these types of updates are useful to solve least squares problems with L_1-norm penalty also referred to as the LASSO [5].

The least absolute shrinkage and selection operator (LASSO), is a shrinkage and selection method for linear regression. It minimizes the usual sum of squared errors, with a bound on the sum of the absolute values also named the L_1-norm of the coefficients [5], i.e.

$$\boldsymbol{\beta} = \arg\min\{\|\mathbf{Y} - \boldsymbol{\beta}\mathbf{X}\|_F^2\} \quad s.t. \quad \sum_m |\boldsymbol{\beta}_m| \leq t, \quad (1)$$

which is equivalent to the minimization

$$\boldsymbol{\beta} = \arg\min\{\frac{1}{2}\|\mathbf{Y} - \boldsymbol{\beta}\mathbf{X}\|_F^2 + \lambda \sum_m |\boldsymbol{\beta}_m|\}. \quad (2)$$

That is, there is a one to one correspondence between t and λ [5, 6]. LASSO has connections to soft-thresholding of wavelet coefficients, forward stagewise regression, and boosting methods [7] and forms a framework to solve the Basis Pursuit [8, 9] with noise (Basis Pursuit Denoising) [6]. The attractive property of the L_1-norm is that it penalizes the non-sparsity of $\boldsymbol{\beta}$ without violating the convexity of the optimization problem. Furthermore, the L_1-norm is known to mimic the behavior of the L_0 norm, i.e. to attain as many zero elements as possible [10] giving the simplest and often also the most parsimonious solution to account for the data.

The equivalent minimization problems given in equation (1) and (2) have been solved by quadratic programming (QP). Since $|\boldsymbol{\beta}_m|$ cannot be handled by regular QP the problem has been recast in the non-negative

variables $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$ such that $\boldsymbol{\beta}_m = \boldsymbol{\beta}_m^+ - \boldsymbol{\beta}_m^-$. Then, the LASSO can be stated in standard QP form by $\widetilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ -\mathbf{X} \end{bmatrix}$ and $\widetilde{\boldsymbol{\beta}} = \left[\boldsymbol{\beta}^+, \boldsymbol{\beta}^-\right]$ subject to the constraint $\widetilde{\boldsymbol{\beta}} \geq \mathbf{0}$. We will currently explore the structure of this reformulated problem to form two algorithms for the LASSO based on multiplicative updates. Using multiplicative updates has the following benefits:

1. The non-negativity constraint of $\widetilde{\boldsymbol{\beta}}$ can naturally be enforced.
2. The fact that $\widetilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ -\mathbf{X} \end{bmatrix}$ can be used to avoid doubling the size of the problem compared to standard QP-solvers.
3. The algorithm based on multiplicative updates is easy to implement, has low computational cost per iteration and is proven to monotonically decrease the cost-function.
4. The multiplicative updates form a general optimization framework which can potentially be used for a wide range of problems.

2. METHOD

Multiplicative updates (MU) were introduced in [2, 3] to solve the non-negative matrix factorization (NMF) which corresponds to

$$\mathbf{Y} \approx \boldsymbol{\beta}\mathbf{X}, \tag{3}$$

where $\mathbf{Y} \in \Re_+^{I\times J}$, $\boldsymbol{\beta} \in \Re_+^{I\times M}$ and $\mathbf{X} \in \Re_+^{M\times J}$ are all non-negative. This was extended to semi-NMF [4] where $\mathbf{Y} \in \Re^{I\times J}$ and $\mathbf{X} \in \Re^{M\times J}$ i.e. for $\boldsymbol{\beta}$ non-negativity constrained while $\mathbf{Y}$ and $\mathbf{X}$ are unconstrained. Given a cost function $C(\boldsymbol{\beta})$ over the non-negative variables $\boldsymbol{\beta}$, define $\frac{\partial C(\boldsymbol{\beta})^+}{\partial \boldsymbol{\beta}_{i,m}}$ and $\frac{\partial C(\boldsymbol{\beta})^-}{\partial \boldsymbol{\beta}_{i,m}}$ as the positive and negative part of the derivative with respect to $\boldsymbol{\beta}_{i,m}$. Then the multiplicative update has the following form

$$\boldsymbol{\beta}_{i,m} \leftarrow \boldsymbol{\beta}_{i,m} \left(\frac{\frac{\partial C(\boldsymbol{\beta})^-}{\partial \boldsymbol{\beta}_{i,m}}}{\frac{\partial C(\boldsymbol{\beta})^+}{\partial \boldsymbol{\beta}_{i,m}}} \right)^{\alpha}. \tag{4}$$

A small constant $\varepsilon = 10^{-9}$ is added to the numerator and denominator to avoid division by zero or forcing $\boldsymbol{\beta}$ to zero. If the gradient is positive $\frac{\partial C(\boldsymbol{\beta})^+}{\partial \boldsymbol{\beta}_{i,m}} > \frac{\partial C(\boldsymbol{\beta})^-}{\partial \boldsymbol{\beta}_{i,m}}$, hence, $\boldsymbol{\beta}_{i,m}$ will decrease and vice versa if the gradient is negative. Thus, there is a one-to-one relation between fixed points and the gradient being zero. α is a "step size" parameter that potentially can be tuned. Notice, when $\alpha \to 0$ only very small steps in the negative gradient direction are taken. The attractive property of multiplicative updates is that they automatically enforce non-negativity while given values of α have been proven to monotonically decrease various cost functions. For NMF the Kullback-Leibler divergence and least squares cost functions are monotonically decreased for $\alpha = 1$ [3] while semi-NMF based on least squares as defined in [4] is monotonically decreased for $\alpha = 0.5$ [4]. Another form of multiplicative updates for semi-NMF is given in [17] derived in the framework of quadratic programming with non-negativity constraints.

Presently, we will demonstrate that multiplicative updates can also be used for unconstrained optimization, that is $\mathbf{Y} \in \Re^{I\times J}$, $\boldsymbol{\beta} \in \Re^{I\times M}$ and $\mathbf{X} \in \Re^{M\times J}$ are unconstrained. Notice, this problem can be trivially solved by matrix inverses. However, it is relevant to solve the problem by multiplicative updates when constraints such as sparseness by the L_1-norm is imposed since a closed form solution no longer exists. Furthermore, such constraints are traditionally imposed when the problem is over complete ($M \gg J$) and matrix inverses become unstable. Without loss of generality we will consider $\boldsymbol{\beta} \in \Re^{1\times M}$. We now have the LASSO problem as stated in equation (2)

$$\boldsymbol{\beta} = \arg\min\{\frac{1}{2}\|\mathbf{Y} - \boldsymbol{\beta}\mathbf{X}\|_F^2 + \lambda \sum_m |\boldsymbol{\beta}_m|\}. \tag{5}$$

If $\boldsymbol{\beta}$ is unconstrained the gradient of the L_1-term, i.e. $P = \lambda \sum_m |\boldsymbol{\beta}_m|$, gives $\frac{\partial P}{\partial \boldsymbol{\beta}} = \lambda \cdot sign(\boldsymbol{\beta})$ ($\boldsymbol{\beta} \neq 0$) such that the contribution from the constraint gives a step of same length regardless of the value of $\boldsymbol{\beta}$. Consequently, for large scale sparse problems oscillations around zero of small elements of $\boldsymbol{\beta}$ makes a simple gradient search get stuck in small step-sizes in order to keep decreasing the cost function. However, by reformulating the problem in the variables $\boldsymbol{\beta}_m = \boldsymbol{\beta}_m^+ - \boldsymbol{\beta}_m^-$ and constraining $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$ to be non-negative elements can no longer cross zero. Furthermore, the non-differentiability at $\boldsymbol{\beta} = 0$ is no longer a concern as $\boldsymbol{\beta}$ only goes to zero from one direction. Presently, non-negativity can naturally be enforced by multiplicative updates. Consider again the reformulated LASSO problem cast in the non-negative variables $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$ to be solvable by QP

$$C_{LASSO} = \frac{1}{2}\|\mathbf{Y} - \widetilde{\boldsymbol{\beta}}\widetilde{\mathbf{X}}\|_F^2 + \lambda \sum_m \widetilde{\boldsymbol{\beta}}_m, \tag{6}$$

where $\widetilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ -\mathbf{X} \end{bmatrix}$ and $\widetilde{\boldsymbol{\beta}} = \left[\boldsymbol{\beta}^+, \boldsymbol{\beta}^-\right]$. The gradient of the cost function is given by

$$\frac{\partial C_{LASSO}}{\partial \widetilde{\boldsymbol{\beta}}} = -(\mathbf{Y} - \widetilde{\boldsymbol{\beta}}\widetilde{\mathbf{X}})\widetilde{\mathbf{X}}^T + \lambda \mathbf{1} \tag{7}$$

Notice further, that

$$\mathbf{Y} - \widetilde{\boldsymbol{\beta}}\widetilde{\mathbf{X}} = \mathbf{Y} - (\boldsymbol{\beta}^+ - \boldsymbol{\beta}^-)\mathbf{X} = \mathbf{Y} - \boldsymbol{\beta}\mathbf{X} \quad (8)$$

$$\mathbf{Y}\widetilde{\mathbf{X}}^T = [\mathbf{Y}\mathbf{X}^T, -\mathbf{Y}\mathbf{X}^T] \quad (9)$$

$$\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^T = \begin{bmatrix} \mathbf{X}\mathbf{X}^T & -\mathbf{X}\mathbf{X}^T \\ -\mathbf{X}\mathbf{X}^T & \mathbf{X}\mathbf{X}^T \end{bmatrix}. \quad (10)$$

Using multiplicative updates (MU) as given in equation (4), we now get (for $\boldsymbol{\beta} \in \Re^{I \times M}$)

$$\boldsymbol{\beta}^+_{i,m} \leftarrow \boldsymbol{\beta}^+_{i,m} \sqrt{\frac{([\mathbf{Y}\mathbf{X}^T]^+ + \boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^- + \boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^+)_{i,m}}{([\mathbf{Y}\mathbf{X}^T]^- + \boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^+ + \boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^-)_{i,m} + \lambda}}$$

$$\boldsymbol{\beta}^-_{i,m} \leftarrow \boldsymbol{\beta}^-_{i,m} \sqrt{\frac{([\mathbf{Y}\mathbf{X}^T]^- + \boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^+ + \boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^-)_{i,m}}{([\mathbf{Y}\mathbf{X}^T]^+ + \boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^- + \boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^+)_{i,m} + \lambda}}$$

where $[\mathbf{M}]^+$ and $[\mathbf{M}]^-$ denotes the positive and negative part of $\mathbf{M}$. Based on the approach of [17] the following multiplicative updates (MUqp) can also be derived

$$\boldsymbol{\beta}^+_{i,m} \leftarrow \boldsymbol{\beta}^+_{i,m} \frac{-\mathbf{P}_{i,m} + \sqrt{\mathbf{P}^2_{i,m} - 4(\boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^+)_{i,m}(\boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^-)_{i,m}}}{2(\boldsymbol{\beta}^+[\mathbf{X}\mathbf{X}^T]^+)_{i,m}}$$

$$\boldsymbol{\beta}^-_{i,m} \leftarrow \boldsymbol{\beta}^-_{i,m} \frac{-\mathbf{R}_{i,m} + \sqrt{\mathbf{R}^2_{i,m} - 4(\boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^+)_{i,m}(\boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^-)_{i,m}}}{2(\boldsymbol{\beta}^-[\mathbf{X}\mathbf{X}^T]^+)_{i,m}}$$

where $\mathbf{P} = -\mathbf{Y}\mathbf{X}^T - \boldsymbol{\beta}^-\mathbf{X}\mathbf{X}^T + \lambda\mathbf{1}$ and $\mathbf{R} = \mathbf{Y}\mathbf{X}^T - \boldsymbol{\beta}^+\mathbf{X}\mathbf{X}^T + \lambda\mathbf{1}$. A proof, that the first type of updates (MU) monotonically decrease the cost function is given in the Appendix, see section 5. An equivalent proof for the second type of updates (MUqp) follows directly from [17]. Thus, the algorithms formed by the updates above do not need to evaluate the objective function. Notice, for both algorithms all that is needed in memory is the precalculated values $\mathbf{Y}\mathbf{X}^T$ and $\mathbf{X}\mathbf{X}^T$ while each iteration requires computations of size $\boldsymbol{\beta}\mathbf{X}\mathbf{X}^T$. Consequently, the computational complexity is given as $\mathcal{O}(IM^2)$. Furthermore, the problem is in theory convex and therefore not prone to local minima. However, one problem is to estimate when the algorithm has converged. Presently, we defined the convergence as a small relative change in $\boldsymbol{\beta}$ less than 10^{-8} or when 20000 iterations had been reached. To speed up the algorithm, we used active sets to disregard very small elements in $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$. Furthermore, for $\lambda = 0$ the activity of $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$ is arbitrary for fixed difference, i.e $\boldsymbol{\beta} = \boldsymbol{\beta}^+ - \boldsymbol{\beta}^-$. Thus, if an element in $\boldsymbol{\beta}$ changed infinitesimally between each iteration the complete activity of this element was placed in either $\boldsymbol{\beta}^+$ or $\boldsymbol{\beta}^-$ depending on the sign of the element in $\boldsymbol{\beta}$ to further reduce the problem size.

3. RESULTS AND DISCUSSION

We tested the two types of multiplicative updates presently derived for the LASSO against the standard solver in Matlab for quadratic programming (QP) and the basis pursuit denoising (BP) algorithm described in [6] which is available from www.sparselab.stanford.edu. Three data sets were considered: One small scale and two large scale problems.

3.1. Small scale data set ($M < J$)

The first example is a well known study performed by [11] also used as an example in [12], where $M < J$. The study examined the correlation between the level of specific prostate antigen and 8 clinical measures ($M = 8$). The clinical measures were taken on 97 men ($J = 97$) who were about to receive a radical prostatectomy.

For the data set, we see that the solutions of MU, MUqp, QP and BP are equivalent in standard error (given as $\sqrt{\frac{1}{J}\sum_{j=1}^{J}(\mathbf{Y}_j - (\widetilde{\boldsymbol{\beta}}\widetilde{\mathbf{X}})_j)^2}$), see figure 1 (a). The cpu-time usage is of same order for MU, MUqp, BP and QP although QP is slightly faster than the other three, see figure 1 (b).

3.2. Large scale data sets ($M \gg J$)

The two large scale data sets consist of microarray data taken from [13] of studies performed by [14] and [15] of colon data and breast cancer data, respectively. The microarrays contain expressions of 2000 and 3226 genes.

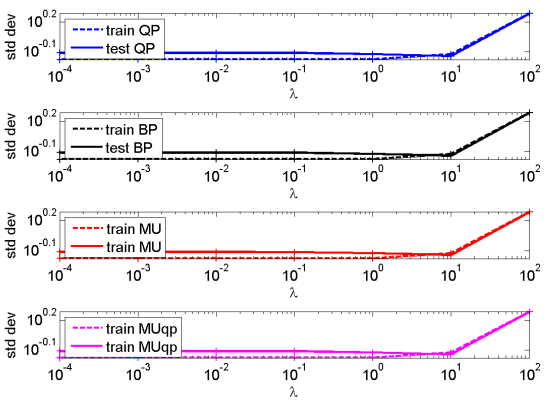
The first data set represents a study of the gene expression for 40 tumor and 22 normal colon tissues. The samples were divided into a training set with 13 normal samples and 27 tumor samples and a test set with 9 normal samples and 13 tumor samples.

The second data set considers gene expressions for carriers of BRCA1 mutation, BRCA2 mutation, and sporadic cases of breast cancer. Here, we will consider the separation of BRCA1 mutations from the tissues with BRCA2 mutations or sporadic mutations. The training set consists of 4 samples with BRCA1 mutations and 10 without. The test set consists of 3 samples with BRCA1 mutations and 5 without.

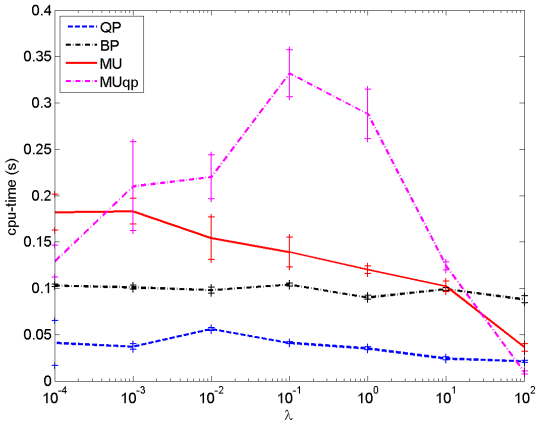
The results obtained from the colon data set as well as the breast cancer data set are given in figure 2 and figure 3, respectively. For small values of λ both MU and MUqp have not fully converged however for large values of λ the solutions are equivalent to BP. Finally, QP is unstable and have problems finding the minima regardless of the values of λ.

4. CONCLUSION AND FUTURE WORK

The present algorithm for the LASSO based on two types of multiplicative updates performed equally well

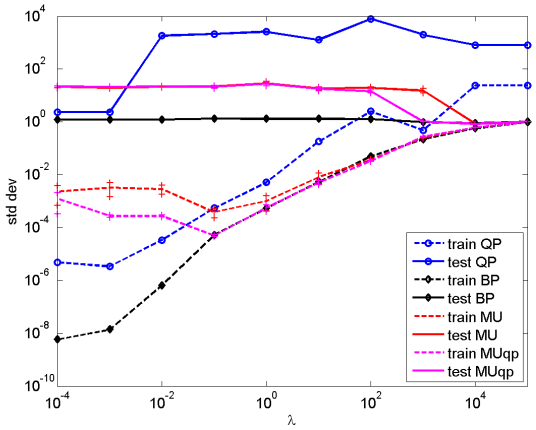


(a) Std dev

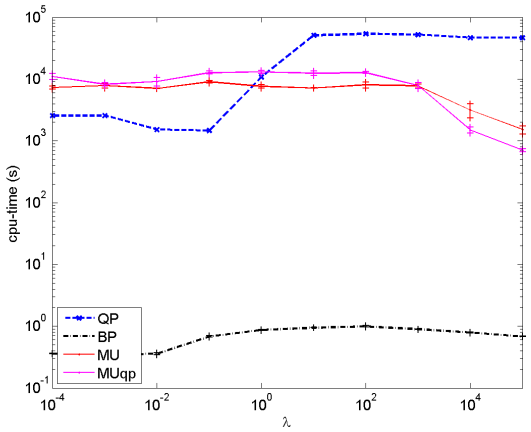


(b) Cpu-time

Fig. 1. The standard deviation and the cpu-time as a function of λ for the prostate cancer data. The solutions found by QP, MU, MUqp and BP are identical while the time-usage is of more or less the same order. The time usage of MU and MUqp reduces for large values of λ due to occurence of zero elements which can be disregarded in the updates. The error bars denotes the standard deviation of the mean of 10 runs.



(a) Std dev

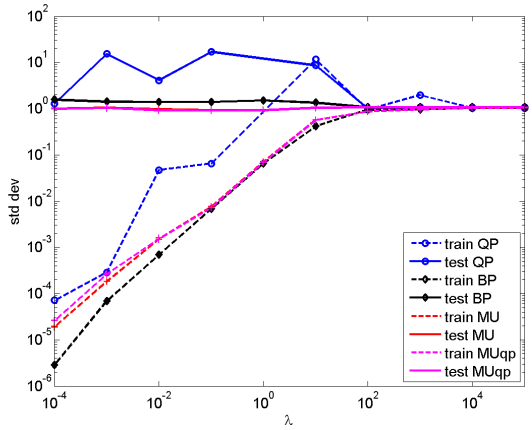


(b) Cpu-time

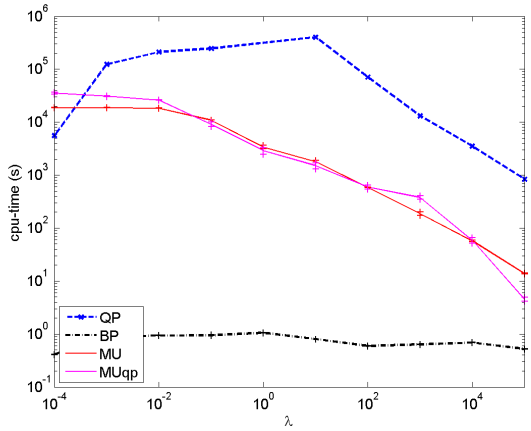
Fig. 2. The standard deviation and the cpu-time as a function of λ for the colon cancer data. While QP is unstable and slow, MU and MUqp are more stable than QP. However, for small values of λ the multiplicative updates are slower than QP and does not fully converge. For large values of λ MU and MUqp is faster than QP and the solutions of MU and MUqp are equivalent to those obtained by BP. The error bars denotes the standard deviation of the mean of 3 runs, due to the large computational time for QP only one run of QP has been included.

for small scale problems as QP and BP. However, for large scale over complete problems BP was much faster than both QP, MU and MUqp. For large values of λ BP, MU and MUqp had same quality of solutions but for low values MU and MUqp did not converge. While QP was unstable for large scale problems this was neither the case for MU, MUqp nor BP. Although, multiplicative updates suffer from slow convergence when λ is small they are simple and easy to implement and clearly outperform QP for large scale problems. However, they are not as good as state of the art algorithms such as the BP algorithm [6]. The present multiplicative updates were based on two different approaches, [4, 17]. Despite their different nature their performances were for the present analysis very similar.

Other algorithms for the LASSO than the present QP and BP exist, see for instance Osborne et al. [16]. Also, algorithms not based on directly minimizing the LASSO cost for a specific value of λ such as least an-



(a) Std dev



(b) Cpu-time

Fig. 3. The standard deviation and the cpu-time as a function of λ for the breast cancer data. The same tendencies are observed as for the colon cancer data in figure 2. Namely, For small values of λ MU and MUqp have not fully converged. Furthermore, QP is again unstable and slow. MU and MUqp is faster and for large values of λ equivalent to BP in quality of solutions obtained. The error bars denotes the standard deviation of the mean of 3 runs, due to the large computational time for QP only one run of QP has been included.

gle regression selection (LARS) [7] and the Homotopy method [18, 16] have recently been proposed. However, these algorithms are based on successively introducing or removing variables rather than directly minimizing the cost-function for a specific value of λ hence do not directly compare to the present algortihms for the LASSO based on multiplicative updates.

The multiplicative updates based on equation (4) is a general framework to solve non-negativity constrained problems and can easily be generalized to other cost-functions and additional constraints. Presently, we demonstrated that multiplicative updates can be used for unconstrained minimization where $\boldsymbol{\beta}$ takes both positive and negative values and how this could be used to form two simple algorithm for the LASSO. Recently, the LASSO has been advanced to the so called "elastic net" which apart from a L_1-norm penalty has an additional L_2-norm penalty on $\boldsymbol{\beta}$. This encourages a grouping effect, where strongly correlated predictors tend to be in or out of the model together and improves the stability in the $M \gg J$ case for small values of λ [19]. Furthermore, the LASSO has been advanced to the fused LASSO where the L_1-norm is imposed on both the coefficients and their successive differences. This encourages local constancy of the coefficient profile and also improves stability in the $M \gg J$ case [20]. It should be possible to advance the present multiplicative updates to both accommodate the elastic net as well as the fused LASSO. This will be the focus of future work. Furthermore, in [21] it was demonstrated that multiplicative updates easily can accommodate missing values - this might be relevant to consider when modeling data using the LASSO. Hence, it is our strong belief that the present multiplicative methods can be extended to form simple algorithms for a wide range of data as well as models.

5. APPENDIX: PROOF OF CONVERGENCE FOR MU $\alpha = 0.5$

The proof is based on the use of an auxiliary function [3] and follows closely the proofs for the convergence of semi-NMF given in [4]. Briefly stated, an auxiliary function G to the function F is defined by: $G(\mathbf{B}, \mathbf{B}') \geq F(\mathbf{B})$ and $G(\mathbf{B}, \mathbf{B}) = F(\mathbf{B})$. If G is an auxiliary function then F is non-increasing under the update $\mathbf{B} = \arg\min_{\mathbf{B}} G(\mathbf{B}, \mathbf{B}')$.

Let $\mathbf{B} \in \Re_+^{I \times M}$. In [4] the following relations are proven to hold

$$
\begin{aligned}
Tr(\mathbf{B}[\mathbf{X}\mathbf{X}^{\mathbf{T}}]^{+}\mathbf{B}) &\leq \sum_{i,m} \frac{([\mathbf{X}\mathbf{X}^T]^{+}\mathbf{B}')_{i,m}\mathbf{B}_{i,m}^2}{\mathbf{B}'_{i,m}} \\
Tr(\mathbf{B}[\mathbf{X}\mathbf{X}^{\mathbf{T}}]^{-}\mathbf{B}) &\geq \sum_{i,m,m'} [\mathbf{X}\mathbf{X}^T]^{-}_{m,m'}\mathbf{B}'_{i,m}\mathbf{B}_{i,m'} \\
&\quad (1 + log\frac{\mathbf{B}_{i,m}\mathbf{B}_{i,m'}}{\mathbf{B}'_{i,m}\mathbf{B}'_{i,m'}}) \\
Tr(\mathbf{B}[\mathbf{Y}]^{-}) &\leq \sum_{i,m} [\mathbf{Y}]^{-}_{i,m} \left(\frac{\mathbf{B}_{i,m}^2 + \mathbf{B}_{i,m}'^2}{2\mathbf{B}'_{i,m}} \right)
\end{aligned}
$$

$$Tr(\mathbf{B}[\mathbf{Y}]^+) \geq \sum_{i,m}[\mathbf{Y}]^+_{i,m}\mathbf{B}_{i,m}(1+\log\frac{\mathbf{B}_{i,m}}{\mathbf{B}'_{i,m}})$$

$$\mathbf{B}_{i,m} \leq \frac{\mathbf{B}^2_{i,m}+\mathbf{B}'^2_{i,m}}{2\mathbf{B}'_{i,m}}$$

The present LASSO costfunction is given as:

$$C(\widetilde{\boldsymbol{\beta}}) = \frac{1}{2}\|\mathbf{Y}-(\boldsymbol{\beta}^+-\boldsymbol{\beta}^-)\mathbf{X}\|^2_F + \lambda\sum_{i,m}(\boldsymbol{\beta}^+_{i,m}+\boldsymbol{\beta}^-_{i,m})$$

$$\begin{aligned} &= \frac{1}{2}Tr(\mathbf{Y}\mathbf{Y}^\mathbf{T}) \\ &+ \frac{1}{2}Tr((\boldsymbol{\beta}^+-\boldsymbol{\beta}^-)([\mathbf{X}\mathbf{X}^T]^+-[\mathbf{X}\mathbf{X}^T]^-)(\boldsymbol{\beta}^+-\boldsymbol{\beta}^-)^T) \\ &- 2Tr((\boldsymbol{\beta}^+-\boldsymbol{\beta}^-)([\mathbf{X}\mathbf{Y}^T]^+-[\mathbf{X}\mathbf{Y}^T]^-)) \\ &+ \lambda\sum_{i,m}(\boldsymbol{\beta}^+_{i,m}+\boldsymbol{\beta}^-_{i,m}) \end{aligned}$$

Using the upper bounds on positive contributions and lower bounds on negative contributions given before, an auxiliary function for $G(\widetilde{\boldsymbol{\beta}},\widetilde{\boldsymbol{\beta}}')$ is derived. Minimizing this function with respect to $\widetilde{\boldsymbol{\beta}}$ we obtain the multiplicative updates with $\alpha = 0.5$.

6. REFERENCES

[1] M. Mørup and L.H. Clemmensen, "Mulasso," http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/5235/zip/imm5235.zip.

[2] D.D. Lee and H.S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–91, 1999.

[3] D.D. Lee and H.S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, 2000, pp. 556–562.

[4] C. Ding, T. Li, and M.I. Jordan, "Convex and semi-nonnegative matrix factorizations," *LBNL Tech Report 60428*, 2006.

[5] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[6] S.S. Chen, D.L. Donoho, and M.A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comp.*, vol. 20, no. 1, pp. 33–61, 1999.

[7] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[8] S.C. Shaobing and D. Donoho, "Basis pursuit," *28th Asilomar conf. Signals, Systems Computers*, 1994.

[9] V. Guigue, A. Rakotomamonjy, and S. Canu, "Kernel basis pursuit," *European Conference on Machine Learning, Porto*, 2005.

[10] D. Donoho, "For most large underdetermined systems of linear equations the minimal l^1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[11] T. Stamey, J. Kabalin, J. McNeal, I. Johnstone, H. Freiha, E. Redwine, and N. Yang, "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate ii. radical prostatectomy treated patients," *Journal of Urology*, vol. 16, pp. 1076–1083, 1989.

[12] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2001.

[13] N. Pochet, F. De Smet, A. K. Suykens, and L. R. De Moor Bart, "Systematic benchmarking of microarray data classification: assessing the role of nonlinearity and dimensionality reduction," *Bioinformatics*, vol. 20, no. 17, pp. 3185–95, 2004.

[14] A. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proc. Natl. Acad. Sci. USA*, 1999.

[15] I. Hedenfalk, D. Duggan, Y. Chen, M. Radmacher, M. Bittner, R. Simon, P. Meltzer, B. Gutsterson, M. Esteller, M. Raffeld, Z. Yakhini, A. Ben-Dor, E. Dougherty, J. Kononen, L. Bubendorf, W. Fehrle, S. Pittaluga, S. Gruvberger, N. Loman, O. Johannsson, H. Olsson, B. Wilfond, G. Sauter, O.-P. Kallioniemi, A. Borg, and J. Trent, "Gene-expression profiles in hereditary breast cancer," *The New England Journal of Medicine*, vol. 344, pp. 539–548, 2001.

[16] M.R. Osborne, B. Presnell, and B.A. Turlach, "A new approach to variable selection in least squares problems," *IMA Journal of Numerical Analysis*, vol. 20, no. 3, pp. 389–403, 2000.

[17] F. Sha, L.K. Saul, and D.D. Lee, "Multiplicative updates for nonnegative quadratic programming in support vector machines," in *Advances in Neural Information Processing Systems 15*, 2002.

[18] I. Drori and D.L. Donoho, "Solution of l1 minimization problems by lars/homotopy methods," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2006.

[19] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Statist. Soc. B*, vol. 67, no. 2, pp. 301–320, 2005.

[20] R. Tibshirani and M.A. Saunders, "Sparsity and smoothness via the fused lasso," *J. R. Statist. Soc. B*, vol. 67, no. 1, pp. 91–108, 2005.

[21] S. Zhang, W. Wang, J. Ford, and F. Makedon, "Learning from incomplete ratings using non-negative matrix factorization," *6th SIAM Conference on Data Mining (SDM)*, 2006.