Problem Statement

We consider the problem of predicting an target variable $y \in \mathbb{Y}$ given an object $\mathbf{x} \in \mathbb{R}^n$. The space $\mathbb{Y} = \{0, 1\}$ for binary classification problem and $\mathbb{Y} = \mathbb{R}$ for regression problem. The goal is to build a model $f(\mathbf{x}|\mathbf{w}), \mathbf{w} \in \mathbb{R}^p$ which outcomes a prediction for each object. There are given the design matrix $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]^\intercal \in \mathbb{R}^{m \times n}$ and the target vector $\mathbf{y} = [y_1, \dots, y_m]^\intercal \in \mathbb{Y}^m$. The goal is to find the optimal weight vector $\mathbf{w}^*$. The weights $\mathbf{w}$ are fitted by the minimization of an error function:

$$\mathbf{w}^* = \arg\min_{\mathbf{w} \in \mathbb{R}^p} S(\mathbf{w}|\mathbf{X}, \mathbf{y}, f). \tag{1}$$

The investigated choices for the error function $S(\mathbf{w}|\mathbf{X}, \mathbf{y}, f)$ are squared error for regression problem:

$$S(\mathbf{w}|\mathbf{X}, \mathbf{y}, f) = \frac{1}{2}\|\mathbf{y} - \mathbf{f}(\mathbf{X}|\mathbf{w})\|_2^2 = \frac{1}{2}\sum_{i=1}^{m} \|y_i - f(\mathbf{x}_i|\mathbf{w})\|^2; \tag{2}$$

cross-entropy for classification problem:

$$S(\mathbf{w}|\mathbf{X}, \mathbf{y}, f) = \sum_{i=1}^{m} \big[y_i \log f(\mathbf{x}_i|\mathbf{w}) + (1 - y_i) \log(1 - f(\mathbf{x}_i|\mathbf{w}))\big]. \tag{3}$$

The number of model weights p could be extremely huge. The problem (1) is solved by iterative optimization procedures. To obtain weights in the next iteration, the updates $\Delta\mathbf{w}$ are added to the current weights

$$\mathbf{w}_k = \mathbf{w}_{k-1} + \Delta\mathbf{w}_{k-1}. \tag{4}$$

This paper suggests to use the Newton optimization procedure to choose weights updates.

Newton method

The Newton method uses the first order optimization condition for the problem (1) and linearize the gradient of $S(\mathbf{w})$ to get the update rule

$$\nabla S(\mathbf{w} + \Delta\mathbf{w}) = \nabla S(\mathbf{w}) + \mathbf{H} \cdot \Delta\mathbf{w},$$

$$\Delta\mathbf{w} = -\mathbf{H}^{-1}\nabla S(\mathbf{w}).$$

where $\mathbf{H} = \nabla^2 S(\mathbf{w})$ is the Hessian matrix of the error function $S(\mathbf{w})$.

In each iteration of the Newton method the update rule (4) is

$$\mathbf{w}_k = \mathbf{w}_{k-1} - \mathbf{H}^{-1}\nabla S(\mathbf{w}).$$

The Newton method is unstable and computationally hard. In each iteration the Hessian matrix should be inverted. It is impossible for singular $\mathbf{H}$.

This paper suggests the robust Newton algorithm. Before the gradient step we propose to select the set of model parameters which have the greatest impact on the error function $S(\mathbf{w})$. The proposed algorithm is implemented to the nonlinear regression problem and binary classification problem.

Nonlinear regression

Assume that the model $f(\mathbf{x}|\mathbf{w})$ is close to linear in the neighborhood of the point $\mathbf{w} + \Delta\mathbf{w}$

$$\mathbf{f}(\mathbf{X}|\mathbf{w} + \Delta\mathbf{w}) \approx \mathbf{f}(\mathbf{X}|\mathbf{w}) + \mathbf{J} \cdot \Delta\mathbf{w},$$

where $\mathbf{J} \in \mathbb{R}^{m \times p}$ is the Jacobian matrix

$$\mathbf{J} = \begin{pmatrix} \frac{\partial f(\mathbf{x}_1|\mathbf{w})}{\partial w_1} & \cdots & \frac{\partial f(\mathbf{x}_1|\mathbf{w})}{\partial w_p} \\ \cdots & \cdots & \cdots \\ \frac{\partial f(\mathbf{x}_m|\mathbf{w})}{\partial w_1} & \cdots & \frac{\partial f(\mathbf{x}_m|\mathbf{w})}{\partial w_p} \end{pmatrix}. \tag{5}$$

Under this assumption the gradient $\nabla S(\mathbf{w})$ and the Hessian matrix $\mathbf{H}$ of the error function (2) equal

$$\nabla S(\mathbf{w}) = \mathbf{J}^\intercal(\mathbf{y} - \mathbf{f}); \quad \mathbf{H} = \mathbf{J}^\intercal\mathbf{J}.$$

It leads to the Gauss-Newton method and the update rule (4) is

$$\mathbf{w}_k = \mathbf{w}_{k-1} + (\mathbf{J}^\intercal\mathbf{J})^{-1}\mathbf{J}^\intercal(\mathbf{f} - \mathbf{y}).$$

The updates $\Delta\mathbf{w}$ is the solution of the linear regression problem

$$\|\mathbf{z} - \mathbf{F}\Delta\mathbf{w}\|_2^2 \to \min_{\Delta\mathbf{w} \in \mathbb{R}^p}, \tag{6}$$

where $\mathbf{z} = \mathbf{f} - \mathbf{y}$ and $\mathbf{F} = \mathbf{J}$.

We consider the feed-forward two layer neural network as the nonlinear model. In this case the model $f(\mathbf{x}|\mathbf{w})$ is given by

$$f(\mathbf{x}|\mathbf{w}) = \sigma(\mathbf{x}^\intercal\mathbf{W}_1)\mathbf{w}_2.$$

Here $\mathbf{W}_1 \in \mathbb{R}^{n \times h}$ the weight matrix which connects the input features with h hidden units, $\sigma(\cdot)$ is a nonlinearity function which applied element-wise, and $\mathbf{w}_2 \in \mathbb{R}^{h \times 1}$ the weight matrix which connects the hidden units with output. The model weight vector $\mathbf{w}$ is a concatenation of vectorized matrices $\mathbf{W}_1$, $\mathbf{w}_2$.

Logistic Regression

For logistic regression problem the model $f(\mathbf{x}|\mathbf{w}) = \text{sigmoid}(\mathbf{x}^\intercal\mathbf{w})$. The gradient and the Hessian of the error function (3) equal

$$\nabla S(\mathbf{w}) = \mathbf{X}^\intercal(\mathbf{f} - \mathbf{y}); \quad \mathbf{H} = \mathbf{X}^\intercal\mathbf{R}\mathbf{X},$$

where $\mathbf{R}$ is a diagonal matrix with $f(\mathbf{x}_i|\mathbf{w}) \cdot (1 - f(\mathbf{x}_i|\mathbf{w}))$ diagonal entries.

The update rule (4) is

$$\mathbf{w}_k = \mathbf{w}_{k-1} + (\mathbf{X}^\intercal\mathbf{R}\mathbf{X})^{-1}\mathbf{X}^\intercal(\mathbf{y} - \mathbf{f}).$$

This algorithm is known as Iteralively Reweighted Least Squares (IRLS) algorithm. The updates $\Delta\mathbf{w}$ is the solution of the linear regression problem

$$\|\mathbf{z} - \mathbf{F}\Delta\mathbf{w}\|_2^2 \to \min_{\Delta\mathbf{w} \in \mathbb{R}^p}, \tag{7}$$

where $\mathbf{z} = \mathbf{R}^{-1/2}(\mathbf{y} - \mathbf{f})$ and $\mathbf{F} = \mathbf{R}^{1/2}\mathbf{X}$.

QPFS

We suggest to find the subset $\mathcal{A} = \{1, \dots p\}$ of the model parameters which we need to optimize in the current optimization step. To find the optimal subset $\mathcal{A}$ we suggest to use the QPFS algorithm. The original algorithm selects features for the linear regression problem

$$\|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 \to \min_{\mathbf{w} \in \mathbb{R}^n}.$$

The goal of the QPFS is to select not correlated features which are relevant to target vector. To formalise this approach let introduce two functions: Sim and Rel. The former measures the redundancy between features, the latter contains relevances between each feature and target vector. We want to minimize the Sim function and maximize the Rel simultaneously.

The QPFS method offers the explicit way to construct the functions Sim and Rel. The method minimizes the following functional

$$\underbrace{\mathbf{a}^\intercal \mathbf{Q} \mathbf{a}}_{\text{Sim}} - \alpha \cdot \underbrace{\mathbf{b}^\intercal \mathbf{a}}_{\text{Rel}} \to \min_{\substack{\mathbf{a} \geqslant 0 \\ \|\mathbf{a}\|_1 = 1}}. \tag{8}$$

The first term is associated with the Sim function and the second with the Rel. The matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ entries measure the pairwise similarities between features. The vector $\mathbf{b} \in \mathbb{R}^n$ expresses the similarities between each feature and the target vector $\mathbf{y}$. The normalized vector $\mathbf{a}$ shows the importance of each feature. This functional penalizes the dependent features by the function Sim and encourages features relevant to the target by the function Rel. The parameter α allows to control the trade-off between the Sim and the Rel terms. The authors of the original QPFS paper suggested the way to select α and make Sim and Rel terms impact are equal

$$\alpha = \frac{\overline{\mathbf{Q}}}{\overline{\mathbf{Q}} + \overline{\mathbf{b}}},$$

where $\overline{\mathbf{Q}}$, $\overline{\mathbf{b}}$ are the mean values of $\mathbf{Q}$ and $\mathbf{b}$ correspondingly.

To find the optimal feature subset the thresholding for $\mathbf{a}$ is applied:

$$j \in \mathcal{A} \Leftrightarrow a_j > \tau.$$

To measure similarity it was proposed to use the absolute value of sample Pearson correlation coefficient or sample mutual information between pairs of features for the Sim function and between features and target vector for the Rel function. The problem 8 is convex if the matrix $\mathbf{Q}$ is positive semidefinite. In general it is not always true. To satisfy this condition we shift the matrix $\mathbf{Q}$ spectrum and replace the matrix $\mathbf{Q}$ by $\mathbf{Q} - \lambda_{\min}\mathbf{I}$, where $\lambda_{\min}$ is a $\mathbf{Q}$ minimal eigenvalue.

Proposal

To make the optimization process of the Newton algorithm is robust and stable we propose to implement the QPFS algorithm to the problems (6) and (7). The QPFS

selects the set $\mathcal{A}$ of weight updates $\Delta\mathbf{w}$ which have the greatest impact to the residuals and pairwise independent.

If vector $\mathbf{z}$ is orthogonal to the columns of the matrix $\mathbf{F}$ the correlation coefficient and mutual information coefficient are equals zero. It leads to the QPFS vector $\mathbf{b} = 0$. We show that for the optimal weights $\mathbf{w}^*$ for nonlinear regression and logistic regression models $\mathbf{F}^\intercal\mathbf{z}$.

- nonlinear regression

$$\mathbf{F}^\intercal\mathbf{z} = \mathbf{J}^\intercal(\mathbf{f} - \mathbf{y}) = -\nabla S(\mathbf{w}^*) = 0.$$

- logistic regression

$$\mathbf{F}^\intercal\mathbf{z} = \mathbf{X}\mathbf{R}^{-1/2}\mathbf{R}^{1/2}(\mathbf{y} - \mathbf{f}) = \mathbf{X}^\intercal(\mathbf{y} - \mathbf{f}) = \nabla S(\mathbf{w}^*) = 0.$$

Step size

The step size of the Newton method could be excessively large. To control the step size of the weight updates we add the parameter t in the update rule (4)

$$\mathbf{w}_k = \mathbf{w}_{k-1} + t\Delta\mathbf{w}_{k-1}; \quad t \in [0, 1].$$

To select the appropriate the step size t the Armijo rule is used. We choose the t as large as possible to satisfy the following condition

$$S(\mathbf{w}_{k-1} + t\Delta\mathbf{w}_{k-1}) < S(\mathbf{w}_{k-1}) + \gamma t\nabla S^\intercal(\mathbf{w}_{k-1})\mathbf{w}_{k-1}; \quad \gamma \in [0, 0.5].$$

Experiment

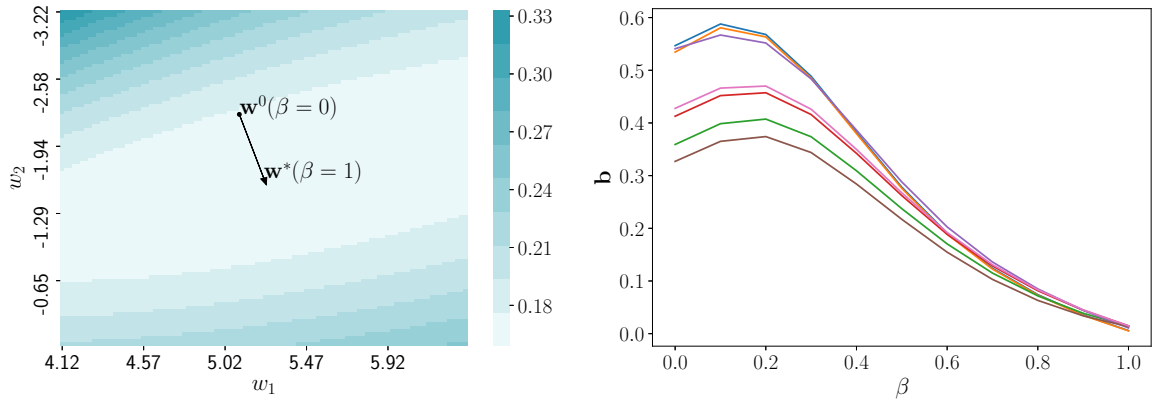
We want to investigate the dependence of the QPFS solution for the problem. Assume that weight vector $\mathbf{w}^0$ lies near the optimal weight vector $\mathbf{w}^*$. We consider the line segment

$$\mathbf{w}_\beta = \beta\mathbf{w}^* + (1 - \beta)\mathbf{w}^0; \; \beta \in [0, 1].$$

In the experiment we used the Boston House Pricing dataset (objects: 506, features: 13).

We fitted the linear model for this data. The error landscape is shown in the figure 3 for two randomly selected weights. We add the random noise to the optimum weights $\mathbf{w}^*$ to get the point $\mathbf{w}^0$. The behaviour of the Rel term vector $\mathbf{b}$ on the line segment between $\mathbf{w}^0$ and $\mathbf{w}^*$ is illustrated in the figure 4.

The landscape for the neural network model is more complex. We use only 2 hidden units to get not excessively complex model. The optimal weight vector $\mathbf{w}^*$ is obtained by backpropagation optimization procedure. Figure 5 shows the error function on the grid of two neural network weights from $\mathbf{W}_1$. We use the same strategy to investigate how the linear term vector $\mathbf{b}$ is changing moving from $\mathbf{w}^0$ to $\mathbf{w}^*$. The result are shown in the figure 6.



(a) Error landscape

(b) Feature relevances

Figure 1: Logistic regression

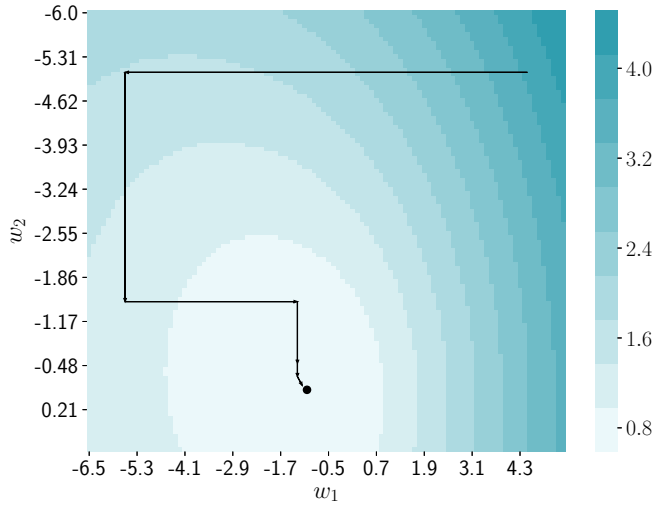


Figure 2: Optimization process for logistic regression with QPFS+Newton algorithm

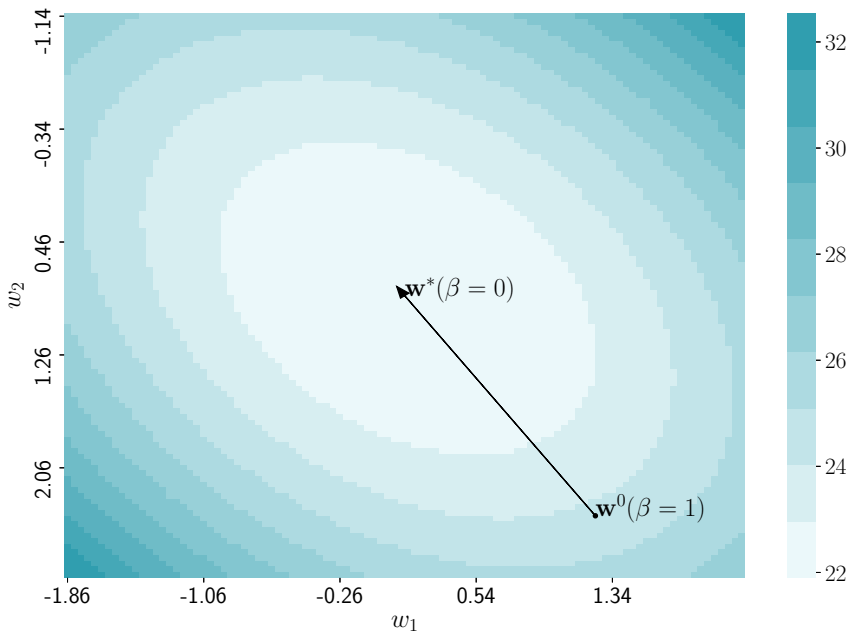


Figure 3: Error function landscape near optimal weight point for linear model.

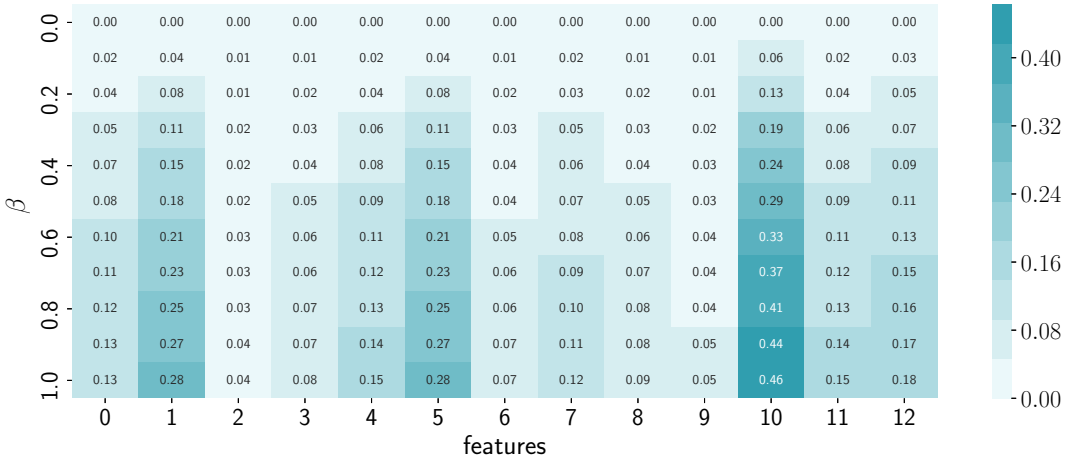


Figure 4: Relevance scores for linear model with respect to β coefficient.

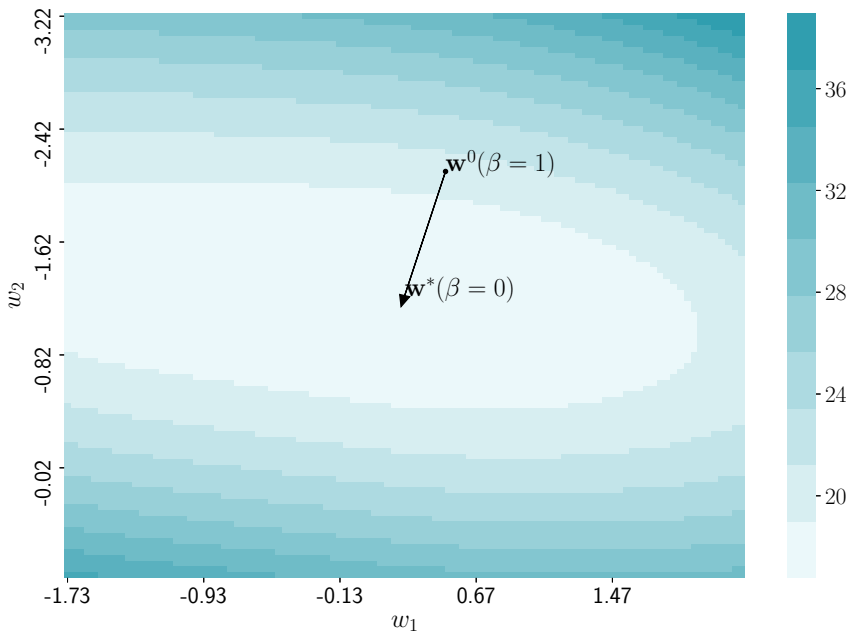


Figure 5: Error function landscape near optimal weight point for neural network.

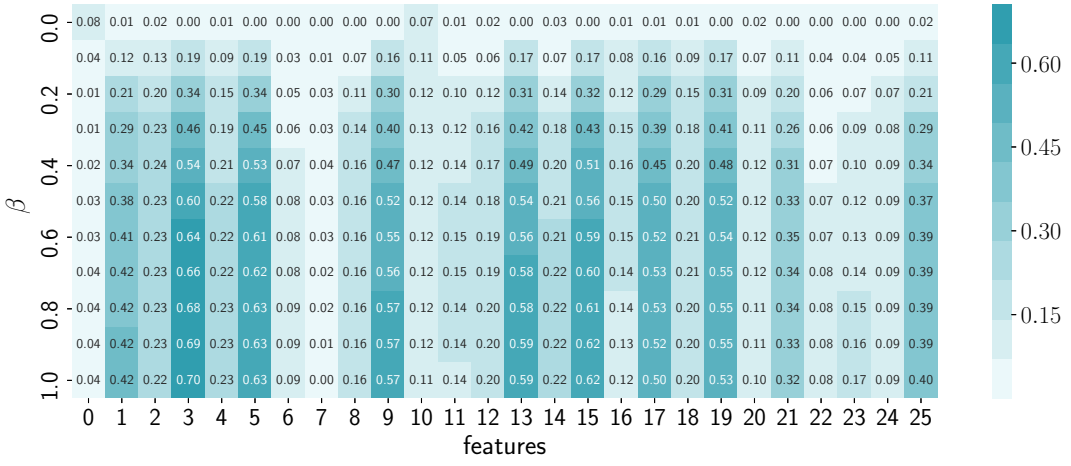


Figure 6: Relevance scores for neural network with respect to β coefficient.