RegERMs Documentation

Release 0.0.1

Christoph Sawade

June 17, 2014

Contents

Regularized empirical risk minimization (RegERM) is a general concept that defines a family of optimization problems in machine learning, as, e.g., Support Vector Machine, Logistic Regression, and Linear Regression.

Contents:

CHAPTER 1

API

The following main concepts are implemented:

1.1 Loss functions

Loss functions measure the disagreement between the true label $y \in \{-1, 1\}$ and the prediction.

Loss functions implement the following main methods:

value (*l::Loss*)
Compute the value of the loss.

gradient (*l::Loss*)
Compute the gradient of the loss.

The following loss functions are implemented:

Logistic (*w::Vector*, *X::Matrix*, *y::Vector*)
Return a vector of the logistic loss evaluated for all given training instances $\mathbf{X}$ and the labels $\mathbf{y}$

$$\ell(\mathbf{w}, \mathbf{x}, y) = \log(1 + exp(-y\mathbf{x}^T\mathbf{w})),$$

where $\mathbf{w}$ is the weight vector of the decision function.

> **Note:** The logistic loss corresponds to a likelihood function under an exponential family assumption of the class-conditional distributions $p(\mathbf{x}|y; \mathbf{w})$.

Squared (*w::Vector*, *X::Matrix*, *y::Vector*)
Return a vector of the squared loss evaluated for all given training instances $\mathbf{X}$ and the labels $\mathbf{y}$

$$\ell(\mathbf{w}, \mathbf{x}, y) = (y - \mathbf{x}^T\mathbf{w})^2,$$

where $\mathbf{w}$ is the weight vector of the decision function.

Hinge (*w::Vector*, *X::Matrix*, *y::Vector*)
Return a vector of the hinge loss evaluated for all given training instances $\mathbf{X}$ and the labels $\mathbf{y}$

$$\ell(\mathbf{w}, \mathbf{x}, y) = \max(0, 1 - y\mathbf{x}^T\mathbf{w}),$$

where $\mathbf{w}$ is the weight vector of the decision function.

> **Note:** The hinge loss corresponds to a max-margin assumption.

1.2 Regularizer

Regularization prevent overfitting and introduce additional information (prior knowledge) to solve an *ill-posed* problem.

Regularizers implement the following main methods:

value (*r::Regularizer*)
Compute the value of the regularizer.

gradient (*r::Regularizer*)
Compute the gradient of the regularizer.

The following regulizers are implemented:

L2reg (*w::Vector*, *λ::Float64*)
Implements an L^2-norm regularization of the weight vector `w` of the decision function:

$$\Omega(\mathbf{w}) = \frac{1}{2\lambda}\|\mathbf{w}\|^2,$$

where λ controls the influence of the regularizer.

Note: The L^2-norm regularization corresponds to Gaussian prior assumption of $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \lambda\mathbf{I})$.

CHAPTER 2

Machine learning methods

The framework implements the following learning algorithms:

2.1 Linear Regression

2.2 Logistic Regression

2.3 Support Vector Machine

Implement: `optimize`

Let $\mathbf{x}_i$ be a vector of features describing an instance i and y_i be its target value. Then, for a given set of n training instances $\{(\mathbf{x}_i$

$$\sum_{i=1}^{n} \ell(\mathbf{w}, \mathbf{x}_i, y_i) + \Omega(\mathbf{w}).$$

The loss function ℓ measures the disagreement between the true label y and the model prediction and the regularizer Ω penalizes the model's complexity.

CHAPTER 3

Indices and tables

- *genindex*
- *modindex*
- *search*