Technische Universität München
Fakultät für Informatik
Prof. N. Navab, Ph.D.

WS 2014
Exercise Sheet 2
October 15, 2014

Practical Course: Machine Learning in Medical Imaging

Linear Classifier and SVM

1 Ordinary Least Squares

1.1 Body fat

The data set `bodyfat` contains several body measurements that can be done using a scale and a tape measure. These can be used to predict the body fat percentage (`body.fat` column). Measuring body fat requires a special apparatus; if our resulting model fits well, we have a low-cost alternative. The measurements are age, weight, height, BMI, neck, chest, abdomen, hip, thigh, knee, ankle, bicep, forearm, and wrist [**?**].

Tasks

1. Create a function `olsfit` which takes a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ of samples and a vector $\mathbf{y} \in \mathbb{R}^n$ of outcomes for each sample. The function should return the *ordinary least squares* (OLS) estimate of the coefficients $\hat{\boldsymbol{\beta}}$ (including the intercept).
2. Create multiple models that predict the amount of body fat based on one of the features mentioned above, respectively. For each model create a scatter plot which depicts the data and the model.
3. Create a single model that contains all of the features mentioned above. Which features have the highest/lowest coefficients?

2 Logistic Regression

2.1 Weighted Least Squares

Weighted Least Squares (WLS) is an extension to least squares, where each sample is associated with a weight w_i. Accordingly, the weighted least squares estimate is defined as

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^T \mathbf{W} \mathbf{X}\right)^{-1} \mathbf{X}^T \mathbf{W}^T \mathbf{y}, \tag{1}$$

where $\mathbf{W}$ is a diagonal matrix containing the weights ($W_{ii} = w_i$).

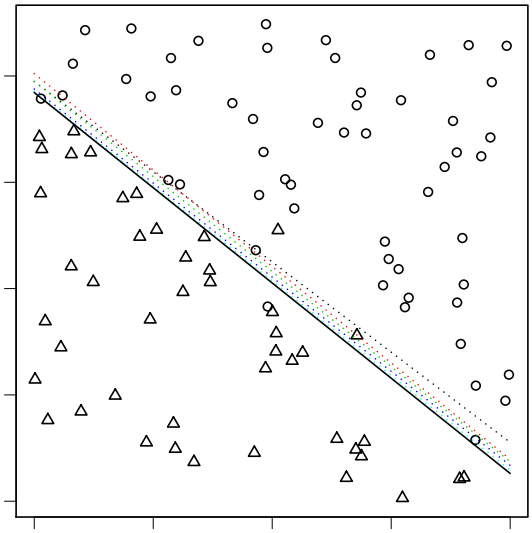

Figure 1: Example of different estimates of coefficients for several iterations of IRLS. The solid line indicates the final fit, dotted lines fits of iterations one to five.

2.2 Iteratively Reweighted Least Squares (IRLS)

The maximum likelihood estimate for logistic regression is defined as

$$\hat{\boldsymbol{\beta}} = \arg\max_{\beta} \sum_{i=1}^{n} y_i \log(\pi_i) + (1 - y_i) \log(1 - \pi_i), \tag{2}$$

where π_i denotes the probability $P(y_i = 1|\mathbf{x}_i)$ defined as

$$P(y_i = 1|\mathbf{x}_i) = \frac{\exp\left(\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}\right)}{1 + \exp\left(\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}\right)}. \tag{3}$$

In comparison to least squares, there is no closed form solution to this problem, therefore one has to find estimates iteratively. A common approach is to iteratively compute a weighted least squares fit until converge, called *iteratively reweighted least squares* (IRLS).

The weights w_i of the diagonal matrix $\mathbf{W}$ are defined as $w_i = \pi_i(1 - \pi_i)$ and the *working response* $\mathbf{z} \in \mathbb{R}^n$ as

$$\mathbf{z} = \mathbf{X}\boldsymbol{\beta}_{\text{old}} + \mathbf{W}^{-1}(\mathbf{y} - \mathbf{p}), \tag{4}$$

where the vector $\mathbf{p} \in \mathbb{R}^n$ contains the fitted probabilities π_i. This leads to the following weighted least squares problem:

$$\boldsymbol{\beta}_{\text{new}} = \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}, \tag{5}$$

where the right side of the equation is evaluated at $\boldsymbol{\beta}_{\text{old}}$.

These equations get solved repeatedly, since at each iteration the vector $\mathbf{p}$ changes, and hence does $\mathbf{W}$ and $\mathbf{z}$. Typically, the convergence criteria is based on the *deviance* ($-2 \cdot$ log-liklihood) and is defined as

$$\frac{|\text{deviance}(\boldsymbol{\beta}_{\text{new}}) - \text{deviance}(\boldsymbol{\beta}_{\text{old}})|}{|\text{deviance}(\boldsymbol{\beta}_{\text{old}})| + 0.1} < 10^{-8} \tag{6}$$

2.3 Likelihood-ratio Test

The likelihood-ratio test can be used to determine if considering additional features in the model results in an increased performance. The test statistic of the likelihood-ratio test D is defined as

$$D = -2 \log \left(\frac{\text{likelihood reduced model}}{\text{likelihood full model}} \right). \tag{7}$$

Under the null-hypothesis that the reduced model performs as well as the full model, D is χ^2 distributed with degrees of freedom (df) equal to the difference in the number of features considered. The resulting p-value gives an indication how likely it is that the result of the likelihood-ratio test arose just from chance. If the p-value is smaller than 0.05, the full model provides a significant benefit over the reduced model.

The test can be applied to assess the significance of all features in a logistic regression model. To obtain a p-value of the i-th feature in the model perform a likelihood-ratio test between the reduced model (i-th feature removed) and the full model. Repeat the process for all features in the model to assess the overall quality of the model.

2.4 South African Heart Disease

The data set `SAheart` is a subset of the Coronary Risk-Factor Study (CORIS) baseline survey, carried out in three rural areas of the Western Cape, South Africa. The aim of the study was to establish the intensity of ischemic heart disease risk factors in that high-incidence region. The data represents white males between 15 and 64, and the response variable is the presence or absence of myocardial infarction (`chd`) at the time of the survey. The data consists of 160 cases, 302 controls and 9 features. The features are systolic blood pressure (`sbp`), cumulative tobacco in kg (`tobacco`), low density lipoprotein cholesterol (`ldl`), adiposity (`adiposity`), family history of heart disease (`famhist`), type-A behaviour (`typea`), obesity (`obesity`), current alcohol consumption (`alcohol`), age at onset (`age`) [**?**, **?**].

Tasks

1. Create a function `lrirlsfit`, which takes a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ of samples and a vector $\mathbf{y} \in \{0; 1\}^n$ of outcomes for each sample. The function should return the *maximum likelihood estimate* (MLE) of the coefficients $\hat{\boldsymbol{\beta}}$ (including the intercept) and the log-likelihood of that model. Use the function `wlsfit` provided by us to compute the WLS fit and initialize $\hat{\boldsymbol{\beta}}$ with all zeros.
2. Create a logistic regression model for the `SAheart` data set.
   a) Create a model that contains only the intercept (**null model**), i.e. no features are considered.
   b) Create multiple models each considering a single feature. Note that `famhist` is a categorical feature which has to be converted to numbers first.

c) Create a function `likelihood_ratio_test` implementing the likelihood-ratio test which takes the log-likelihood of the full model and the reduced model. The function should return the p-value and the test statistic D of the likelihood-ratio test. The p-value can be calculated using the call `gammainc(D/2, df/2, 'upper')`. Use this function to compare the *single feature models* to the *null model*. Which feature yields the most significant improvement over the null model?

d) What do the estimated coefficients tell with respect to the odds of suffering from myocardial infarction? Make sure you consider the p-value of the likelihood-ratio test as well.

e) Create a model which considers multiple features by starting with the null model and adding one additional feature at a time. To determine which feature to add, use the p-value as returned by the likelihood-ratio test. Extended models with one additional feature, where the p-value is greater than 0.05, should not be considered. In each step choose the model with the smallest p-value. Continue until all features have been selected or the model cannot be improved significantly any more.

3 Kernel Ridge Regression

Ridge regression is an extension to ordinary least squares by adding a regularization term to the loss function. It is defined as

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 + \lambda \|\boldsymbol{\beta}\|_2^2, \tag{8}$$

where the value of $\lambda > 0$ determines the amount of regularization. By replacing $\boldsymbol{\beta}$ with $\sum_{i=1}^{n} \alpha_i \mathbf{x}_i$ we obtain

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j \mathbf{x}_i^T \mathbf{x}_j \right)^2 + \lambda \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j \tag{9}$$

As in support vector machines, we can use the Kernel trick to make ridge regression non-linear and at the same time avoid explicitly transforming features. By specifying $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}')$, we obtain the objective function of Kernel Ridge Regression:

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \right)^2 + \lambda \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{10}$$

Estimation of $\boldsymbol{\alpha}$ can still be achieved in closed form with $\hat{\boldsymbol{\alpha}} = (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y}$, where $\mathbf{K}$ is the Kernel matrix with $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. The decision function then becomes

$$f(\mathbf{x}_0) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_0). \tag{11}$$

Tasks

1. Create a function `kernel_ridge_fit`, which takes a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ of samples, a vector $\mathbf{y} \in \mathbb{R}^n$ of outcomes for each sample, and a *kernel function*. `kernel_ridge_fit` should return the estimated vector $\boldsymbol{\alpha} \in \mathbb{R}^n$ and the *mean squared error* on the training samples.
2. Create a function `kernel_ridge_predict`, which takes the matrix $\mathbf{X}$ used during training, the vector $\boldsymbol{\alpha}$, a matrix $\mathbf{X}_t$ of samples for testing and their respective outcomes $\mathbf{y}_t$, and a *kernel function*. The function should return a vector of predicted outcomes and the *mean squared error* on the testing samples.
3. Apply your implementation of kernel ridge regression to the `Aqua-all.csv` data set. The first column denotes the outcome $\mathbf{y}$, the remaining columns the features.
   a) Plot the mean squared *training error* of the first 100 samples obtained from `kernel_ridge_fit` for $\lambda \in \{0.001, 0.01, 0.1, 1, 10, 20, 50, 100, 200, 500, 1000, 10000\}$ and the following Kernel functions: linear, polynomial ($c = 1$, $d = 2$), sigmoid ($\gamma = -0.001$, $c = 1$), and RBF ($\gamma = 0.001$). The x axis should denote the λ values in log scale and the y axis the mean squared error.
   b) Use the remaining 97 samples to test all models trained above and plot the mean squared *test error* obtained from `kernel_ridge_predict`. The x-axis should denote the λ values in log scale and the y-axis the mean squared error.

References

[1] John Verzani. *Using R for Introductory Statistics*, page 296. Chapman and Hall, 2004.

[2] J. Rousseauw, J. du Plessis, A. Benade, P. Jordaan, J. Kotze, and J Ferreira. Coronary risk factor screening in three rural communities. *South African Medical Journal*, 64:430–436, 1983.

[3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*, pages 122–124. Springer, second edition, 2009.