Seon Goh Ann Ray
G2202190G

AI6012: Machine Learning Methodologies & Applications Assignment (25 points)

Important notes: to finish this assignment, you are allowed to look up textbooks or search materials via Google for reference. NO plagiarism from classmates is allowed.

The file to be submitted is a single PDF (no source codes are required to be submitted). Multiple submission attempts are allowed, and the last one will be graded. A submission link is available under "Assignments" of the course website in NTULearn.

Submit to TA!

Question 1 (10 marks): Consider a multi-class classification problem of C classes. Based on the parametric forms of the conditional probabilities of each class introduced on the 39th Page ("Extension to Multiple Classes") of the lecture notes of L4, derive the learning procedure of **regularized** logistic regression for multi-class classification problems.

Hint: define a loss function by borrowing an idea from binary classification, and derive the gradient descent rules to update $\{\mathbf{w}^{(\mathbf{c})}\}$'s.

Question 2 (5 marks): This is a hands-on exercise to use the SVC API of scikit-learn[1] to train a SVM with the linear kernel and the rbf kernel, respectively, on a binary classification dataset. The details of instructions are described as follows.

1. Download the a9a dataset from the LIBSVM Dataset page.

   This is a preprocessed dataset of the Adult dataset in the UCI Irvine Machine Learning Repository[2], which consists of a training set (available here) and a test set (available here).

   Each file (the train set or the test set) is a text format in which each line represents a labeled data instance as follows:

   ```
   label  index1:value1  index2:value2  ...
   ```

   where "label" denotes the class label of each instance, "indexT" denotes the T-th feature, and valueT denotes the value of the T-th feature of the instance.

[1] Read Pages 63-64 of the lecture notes of L5 for reference

[2] The details of the original Adult dataset can be found here.

This is a sparse format, where only non-zero feature values are stored for each instance. For example, suppose given a data set, where each data instance has 5 dimensions (features). If a data instance whose label is "+1" and the input data instance vector is [2 0 2.5 4.3 0], then it is presented in a line as

```
+1 1:2 3:2.5 4:4.3
```

Hint: sciki-learn provides an API ("sklearn.datasets.load_svmlight_file") to load such a sparse data format. Detailed information is available here.

2. Regarding the linear kernel, show 3-fold cross-validation results in terms of classification accuracy on the training set with different values of the parameter C in $\{0.01, 0.05, 0.1, 0.5, 1\}$, respectively, in the following table. Note that for all the other parameters, you can simply use the default values or specify the specific values you used in your submitted PDF file.

Table 1: The 3-fold cross-validation results of varying values of C in SVC with linear kernel on the a9a training set (in accuracy).

$C = 0.01$	$C = 0.05$	$C = 0.1$	$C = 0.5$	$C = 1$
?	?	?	?	?

3. Regarding the rbf kernel, show 3-fold cross-validation results in terms of classification accuracy on the training set with different values of the parameter $gamma$ (i.e., σ^2 on the lecture notes) in $\{0.01, 0.05, 0.1, 0.5, 1\}$ and different values of the parameter C in $\{0.01, 0.05, 0.1, 0.5, 1\}$, respectively, in the following table. Note that for all the other parameters, you can simply use the default values or specify the specific values you used in your submitted PDF file.

Table 2: The 3-fold cross-validation results of varying values of $gamma$ and C in SVC with rbf kernel on the a9a training set (in accuracy).

	$g = 0.01$	$g = 0.05$	$g = 0.1$	$g = 0.5$	$g = 1$
$C = 0.01$	?	?	?	?	?
$C = 0.05$	?	?	?	?	?
$C = 0.1$	?	?	?	?	?
$C = 0.5$	?	?	?	?	?
$C = 1$	?	?	?	?	?

Hint: there are no specific APIs that integrates cross-validation into SVMs in sciki-learn. However, you can use some APIs under the category "Model Selection $\rightarrow$ Model validation" to implement it. Some examples can be found here.

4. Based on the results shown in Tables 1,2, determine the best kernel and the best parameter setting. Use the best kernel with the best parameter setting to train a SVM using the whole training set and make predictions on test set to generate the following table:

Table 3: Test results of SVC on the a9a test set (in accuracy).

	Specify which kernel with what parameter setting
Accuracy of SVMs	?

Question 3 (5 marks): The optimization problem of linear soft-margin SVMs can be re-formulated as an instance of empirical structural risk minimization (refer to Page 37 on L5 notes). Show how to reformulate it. Hint: search reference about the hinge loss.

Question 4 (5 marks): Using the kernel trick introduced in L5 to extend the regularized linear regression model (L3) to solve nonlinear regression problems. Derive a closed-form solution (i.e., to derive a kernelized version of the closed-form solution on Page 50 of L3).

Question 1 (10 marks): Consider a multi-class classification problem of C classes. Based on the parametric forms of the conditional probabilities of each class introduced on the 39th Page ("Extension to Multiple Classes") of the lecture notes of L4, derive the learning procedure of **regularized** logistic regression for multi-class classification problems.

Hint: define a loss function by borrowing an idea from binary classification, and derive the gradient descent rules to update $\{\mathbf{w}^{(c)}\}$'s.

$$P(y=c|x)_i = \frac{e^{-w^{(c)T}x_i}}{1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}} = \hat{y}_{ci} \quad \text{for } c>0,\ i=1,2\ldots N$$

$$P(y=0|x)_i = \frac{1}{1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}} = \hat{y}_{0i} \quad \text{for } c=0,\ i=1,2\ldots N$$

Loss function for binary classification:

$$L(\hat{y},y) = \sum_{i=1}^{N}\left[y_i \ln(\hat{y}_i) + (1-y_i)\ln(1-\hat{y}_i)\right]$$

Loss function for multi-class classification:

$$L(\hat{y},y) = \sum_{i=1}^{N}\left[y_{0i}\ln(\hat{y}_{0i}) + \sum_{c=1}^{C-1} y_{ci}\ln(\hat{y}_{ci})\right]$$

Gradient descent:

$$w_{t+1} = w_t - \rho\frac{\partial E(w)}{\partial w}, \text{ whereby}$$

$$E(w) = -\sum_{i=1}^{N}\left[y_{0i}\ln(\hat{y}_{0i}) + \sum_{c=1}^{C-1} y_{ci}\ln(\hat{y}_{ci})\right] + \frac{\lambda}{2}\|w\|_2^2$$

$$\frac{\partial E(w)}{\partial w} = -\sum_{i=1}^{N}\left[\underbrace{y_{0i}\frac{\partial \ln(\hat{y}_{0i})}{\partial w}}_{1} + \underbrace{\sum_{c=1}^{C-1} y_{ci}\frac{\partial \ln(\hat{y}_{ci})}{\partial w}}_{2}\right] + \lambda w$$

①:

$$y_{oi}\frac{\partial \ln(\hat{y}_{oi})}{\partial w} = y_{oi}\left(\frac{1}{\hat{y}_{oi}}\right)\frac{\partial}{\partial w}\left(\hat{y}_{oi}\right)$$

$$=\left(\frac{y_{oi}}{\hat{y}_{oi}}\right)\frac{\partial}{\partial w}\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}\right)^{-1}$$

$$=\left(\frac{y_{oi}}{\hat{y}_{oi}}\right)(-1)\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}\right)^{-2}\left(0-x_i e^{-w^{(c)T}x_i}\right)$$

$$=\left(\frac{x_i y_{oi}}{\hat{y}_{0i}}\right)\left(\frac{e^{-w^{(c)T}x_i}}{1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}}\right)\left(\frac{1}{1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}}\right)$$

$$=\left(\frac{x_i y_{oi}}{\hat{y}_{oi}}\right)\left(\hat{y}_{ci}\right)\left(\hat{y}_{oi}\right)$$

$$= x_i y_{oi}\hat{y}_{ci}$$

(2):

$$\sum_{c=1}^{C-1} y_{ci} \frac{\partial \ln(\hat{y}_{ci})}{\partial w} = \sum_{c=1}^{C-1} y_{ci} \left(\frac{1}{\hat{y}_{ci}}\right) \frac{\partial}{\partial w}\left(\hat{y}_{ci}\right)$$

$$= \sum_{c=1}^{C-1} \left(\frac{y_{ci}}{\hat{y}_{ci}}\right) \frac{\partial}{\partial w}\left(\frac{e^{-w^{(c)T}x_i}}{1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}}\right)$$

$$= \sum_{c=1}^{C-1} \left(\frac{y_{ci}}{\hat{y}_{ci}}\right) \left[\frac{\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}\right)\left(-x_i e^{-w^{(c)T}x_i}\right) - \left(e^{-w^{(c)T}x_i}\right)\left(0 - x_i e^{-w^{(c)T}x_i}\right)}{\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i}\right)^2}\right]$$

$$= \sum_{c=1}^{C-1} \left(\frac{-x_i y_{ci}}{\hat{y}_{ci}}\right) \left[\frac{\left(e^{-w^{(c)T}x_i}\right)\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x_i} - e^{-w^{(c)T}x_i}\right)}{\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x}\right)\left(1+\sum_{c=1}^{C-1} e^{-w^{(c)T}x}\right)}\right]$$

$$= \sum_{c=1}^{C-1} \left(\frac{-x_i y_{ci}}{\hat{y}_{ci}}\right) \left[\left(\hat{y}_{ci}\right)\left(1 - \hat{y}_{ci}\right)\right]$$

$$= \sum_{c=1}^{C-1} -x_i y_{ci}\left(1 - \hat{y}_{ci}\right)$$

$$= x_i \sum_{c=1}^{C-1} y_{ci}\left(\hat{y}_{ci} - 1\right)$$

$$\therefore \frac{\partial E(w)}{\partial w} = -\sum_{i=1}^{N}\left[x_i y_{0i}\hat{y}_{ci} + x_i \sum_{c=1}^{C-1} y_{ci}(\hat{y}_{ci} - 1)\right] + \lambda w$$

$$= -\sum_{i=1}^{N}\left[x_i\left(y_{0i}\hat{y}_{ci} + \sum_{c=1}^{C-1} y_{ci}\hat{y}_{ci} - \sum_{c=1}^{C-1} y_{ci}\right)\right] + \lambda w$$

$$= -\sum_{i=1}^{N}\left[x_i\left(\sum_{c=0}^{C-1} y_{ci}\hat{y}_{ci} - \sum_{c=1}^{C-1} y_{ci}\right)\right] + \lambda w$$

$$= -\sum_{i=1}^{N}\left[x_i\left(\hat{y}_{ci} - y_{ci}\right)\right] + \lambda w$$

$$\therefore w_{t+1} = w_t - \rho \frac{\partial E(w)}{\partial w}$$

$$= w_t - \rho\left[-\sum_{i=1}^{N} x_i\left(\hat{y}_{ci} - y_{ci}\right) + \lambda w\right]$$

$$= w_t + \rho\left[\sum_{i=1}^{N} x_i\left(\hat{y}_{ci} - y_{ci}\right) - \lambda w\right]$$

Question 2 (5 marks): This is a hands-on exercise to use the SVC API of scikit-learn[1] to train a SVM with the linear kernel and the rbf kernel, respectively, on a binary classification dataset. The details of instructions are described as follows.

1. Download the a9a dataset from the LIBSVM Dataset page.

   This is a preprocessed dataset of the Adult dataset in the UCI Irvine Machine Learning Repository[2], which consists of a training set (available here) and a test set (available here).

   Each file (the train set or the test set) is a text format in which each line represents a labeled data instance as follows:

label index1:value1 index2:value2 ...

   where "label" denotes the class label of each instance, "indexT" denotes the T-th feature, and valueT denotes the value of the T-th feature of the instance.

[1] Read Pages 63-64 of the lecture notes of L5 for reference

[2] The details of the original Adult dataset can be found here.

   This is a sparse format, where only non-zero feature values are stored for each instance. For example, suppose given a data set, where each data instance has 5 dimensions (features). If a data instance whose label is "+1" and the input data instance vector is [2 0 2.5 4.3 0], then it is presented in a line as

+1 1:2 3:2.5 4:4.3

   Hint: sciki-learn provides an API ("sklearn.datasets.load_svmlight_file") to load such a sparse data format. Detailed information is available here.

2. Regarding the linear kernel, show 3-fold cross-validation results in terms of classification accuracy on the training set with different values of the parameter C in $\{0.01, 0.05, 0.1, 0.5, 1\}$, respectively, in the following table. Note that for all the other parameters, you can simply use the default values or specify the specific values you used in your submitted PDF file.

Table 1: The 3-fold cross-validation results of varying values of C in SVC with linear kernel on the a9a training set (in accuracy).

$C = 0.01$	$C = 0.05$	$C = 0.1$	$C = 0.5$	$C = 1$
?	?	?	?	?

3. Regarding the rbf kernel, show 3-fold cross-validation results in terms of classification accuracy on the training set with different values of the parameter $gamma$ (i.e., σ^2 on the lecture notes) in $\{0.01, 0.05, 0.1, 0.5, 1\}$ and different values of the parameter C in $\{0.01, 0.05, 0.1, 0.5, 1\}$, respectively, in the following table. Note that for all the other parameters, you can simply use the default values or specify the specific values you used in your submitted PDF file.

Table 2: The 3-fold cross-validation results of varying values of $gamma$ and C in SVC with rbf kernel on the a9a training set (in accuracy).

	$g = 0.01$	$g = 0.05$	$g = 0.1$	$g = 0.5$	$g = 1$
$C = 0.01$	?	?	?	?	?
$C = 0.05$	?	?	?	?	?
$C = 0.1$	?	?	?	?	?
$C = 0.5$	?	?	?	?	?
$C = 1$	?	?	?	?	?

   Hint: there are no specific APIs that integrates cross-validation into SVMs in sciki-learn. However, you can use some APIs under the category "Model Selection → Model validation" to implement it. Some examples can be found here.

4. Based on the results shown in Tables 1-2, determine the best kernel and the best parameter setting. Use the best kernel with the best parameter setting to train a SVM using the whole training set and make predictions on test set to generate the following table:

Table 3: Test results of SVC on the a9a test set (in accuracy).

	Specify which kernel with what parameter setting
Accuracy of SVMs	?

Table 1: SVC, Linear kernel

C=0.01	C=0.05	C=0.1	C=0.5	C=1
0.844016	0.846104	0.846442	0.846934	0.847210

Table 2: SVC, RBF kernel

	g=0.01	g=0.05	g=0.1	g=0.5	g=1
C=0.01	0.759190	0.819907	0.819846	0.759190	0.759190
C=0.05	0.831209	0.835755	0.834250	0.789165	0.759190
C=0.1	0.837720	0.839655	0.838764	0.806118	0.761985
C=0.5	0.842972	0.845766	0.846811	0.832161	0.789748
C=1	0.844415	0.846749	0.847425	0.836614	0.798286

Table 3: SVC, RBF kernel, C=1, gamma=0.1

	RBF kernel, C=1, gamma=0.1
Accuracy on Test dataset	0.850316

Question 3 (5 marks): The optimization problem of linear soft-margin SVMs can be re-formulated as an instance of empirical structural risk minimization (refer to Page 37 on L5 notes). Show how to reformulate it. Hint: search reference about the hinge loss.

Linear soft-margin SVM:

$$① : \min_{w,b,\xi_i} \frac{\|w\|_2^2}{2} + C\left(\sum_{i=1}^{N} \xi_i\right) \quad \text{s.t. } \begin{array}{l} y_i(w \cdot x_i + b) \geq 1 - \xi_i \\ i = 1, 2, \dots, N \\ \xi_i \geq 0 \end{array}$$

Since $y_i(w \cdot x_i + b) \geq 1 - \xi_i$,

$$\xi \geq 1 - y_i(w \cdot x_i + b)$$

Since we want to minimize ①, we let $\xi = 1 - y_i(w \cdot x_i + b)$

$$\therefore \text{New } ①: \min_{w,b} \frac{\|w\|_2^2}{2} + C \sum_{i=1}^{N} \left[1 - y_i(w \cdot x_i + b)\right]$$

Empirical structural risk minimization:

$$② : \min \underbrace{\sum_{i=1}^{N} l(f(x_i, \theta), y_i)}_{③} + \underbrace{\lambda \Omega(\theta)}_{④}$$

By setting the loss function in ③:

$$l[f(x_i, \theta), y_i] = 1 - y_i(\theta \cdot x_i + b)$$

By setting the λ and regularization term in ④:

$$\lambda \Omega(\theta) = \frac{1}{2} \|\theta\|_2^2 \qquad (\text{L2 norm})$$

$$\text{New } ②: \min_{\theta, b} \sum_{i=1}^{N} \left[1 - y_i(\theta \cdot x_i + b)\right] + \frac{\|\theta\|_2^2}{2}$$

Since the C in new ① is a constant trade-off parameter, the optimization problem of new ① is equivalent to that of new ②

Question 4 (5 marks): Using the kernel trick introduced in L5 to extend the regularized linear regression model (L3) to solve nonlinear regression problems. Derive a closed-form solution (i.e., to derive a kernelized version of the closed-form solution on Page 50 of L3).

Regularized linear regression model:

$$\hat{w} = \min_{w} \frac{1}{2} \sum_{i=1}^{N} (w \cdot x_i - y_i)^2 + \frac{\lambda}{2} \|w\|_2^2$$

$$\sum_{i=1}^{N} w \cdot x_i = \sum_{i=1}^{N} \beta_i x_i \quad , \text{ where } \beta_i \text{ is a scalar}$$

Using kernel trick & dual-form:

$$\hat{\beta} = \min_{\beta} \frac{1}{2} \sum_{i=1}^{N} \left(\sum_{j=1}^{N} \beta_j k(x_i, x_j) - y_i \right)^2 + \frac{\lambda}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\beta_i \beta_j k(x_i, x_j) \right)$$

$$= \min_{\beta} \frac{1}{2} \left(\beta^T k^T k \beta - 2 \beta^T k^T y + y^T y \right) + \frac{\lambda}{2} \left(\beta^T k \beta \right)$$

Set gradient of $\hat{\beta}$ to zero:

$$\frac{\partial \hat{\beta}}{\partial \beta} \Rightarrow \frac{1}{2} \left(2 k^T k \beta - 2 k^T y + 0 \right) + \frac{\lambda}{2} \left(k \beta + k^T \beta \right) = 0$$

$$2 k^T k \beta - 2 k^T y + \lambda k \beta + \lambda k^T \beta = 0$$

$$(2 k^T k + \lambda k + \lambda k^T) \beta = 2 k^T y$$

$$\therefore \beta = 2 (2 k^T k + \lambda k + \lambda k^T)^{-1} k^T y$$