Chapter 12

Logistic Regression and Machine Learning

12.1 Introduction to Machine Learning

- scalable statistical algorithms that combine

  ★ expertise from statistics on how to extract information from data with

  ★ computational ideas that enable efficient implementation on large data sets.

Data revolution: Enormous datasets are routinely collected

- Biological Sci.: Genomics, genetics, neuroscience, medicine
• **Natural Sci.**: Astronomy, earth sciences, meteorology.

• **Engineering**: Machine learning, surveil. videos, social media

• **Social Sci**: Economics, finance, marketing, managements.

Characterize many contemporary scientific and decision problems.
**Example 12.1** *Supervised learning — classification*

**Labels are provided.** Document classification, disease classification, face recognition.

**Feature Space** $\mathcal{X}$
- Words in a document
- Cell properties

**Label Space** $\mathcal{Y}$
- “Sports”
- “News”
- “Science”
- “Anemic cell”
- “Healthy cell”

Figure 12.1: Some supervised learning problems
Example 12.2 Unsupervised learning — clustering

No labels provided: Social network and animal phylogenetic tree

Example 12.3 Gene expression and autism

Over 60K gene expression profiles (Next Generation Sequencing) are measured among 104 samples: 47 autisms and 57 healthy controls, along with gender, brain region, age, and sites. Of interest is to find the genes that are associated with autism.
12.2 Logistic regression

**Modeling binary data**: Suppose that latent variable (e.g. severity of disease such as autism and cancers) follows

\[ Z = \beta_0^* + \beta_1 X_1 + \cdots + \beta_k X_k + \varepsilon = \beta_0^* + \beta^T X + \varepsilon, \]

where \( \beta = (\beta_1, \cdots, \beta_k)^T \) and \( X = (X_1, \cdots, X_k)^T \). Instead of observing \( Z \), we get \( Y = I(Z < c) \) for a threshold \( c \).

**Conditional probability**: if \( \varepsilon \sim G \)

\[
P(Y = 1|X = x^*) = P(\beta_0^* + x^T \beta + \varepsilon < c)
= G(c - \beta_0^* - x^T \beta)
= F(\beta_0 + x^T \beta)
\]
where $F(x) = G(-x)$ and $\beta_0 = \beta_0^* - c$.

**Link function**: $F^{-1}(\cdot)$ is called link. Commonly used examples:

- **logit link**: $F(x) = \frac{\exp(x)}{1 + \exp(x)}$, $F^{-1}(p) = \log \frac{p}{1-p}$ — logit function
  
  $$P(Y = 1 | X = x^*) = \frac{\exp(\beta_0 + x^* T \beta)}{1 + \exp(\beta_0 + x^* T \beta)},$$

- **probit link**: $F(x) = \Phi(x)$, normal cdf.

**Observed data**: $\{(x_i, y_i)\}_{i=1}^n$, $y_i = \text{binary}$.

**pmf for Bernoulli**: $P(Y = y) = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{if } y = 0 \end{cases} = pq^{1-y}$. 

**MLE**: Find $\beta_0$ and $\beta$ to maximize

$$L(\beta_0, \beta) = \prod_{i=1}^{n} p_i^{y_i} q_i^{1-y_i}$$

where $p_i = F(\beta_0 + x_i^T \beta)$. Its log-likelihood is

$$\ell(\beta_0, \beta) = \sum_{i=1}^{n} y_i \log(p_i/q_i) + \log q_i$$

**Logistic regression**: $p_i = \frac{\exp(\beta_0 + x_i \beta)}{1 + \exp(\beta_0 + x_i \beta)}$ and $q_i = \frac{1}{1 + \exp(\beta_0 + x_i \beta)}$ (logit link). Find $\beta_0$ and $\beta$ to maximize

$$\ell(\beta_0, \beta) = \sum_{i=1}^{n} y_i (\beta_0 + x_i \beta) - \log(1 + \exp(\beta_0 + x_i \beta))$$.
Solution: $\beta_0$ and $\hat{\beta} = (\hat{\beta}_1, \cdots, \hat{\beta}_k)'$, by convex optimization.

Predicted probability: $P(Y = 1|x = x^*) = F(\hat{\beta}_0 + x^*\hat{\beta})$.

Example 12.4 Sex classification using heights

User profile data for 59,946 San Francisco OkCupid users (a free online dating website) from June 2012 are recorded.

```r
logistic.model <- glm(is.female ~ height, family=binomial, data=profiles)
summary(logistic.model)
```

Coefficients:

|          | Estimate | Std. Error | z value | Pr(>|z|) |
|----------|----------|------------|---------|----------|
| (Intercept) | 44.9999 | 1.1374 | 39.6 | <2e-16 *** |
| height   | -0.6705 | 0.0169 | -39.8 | <2e-16 *** |

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 8075.1 on 5994 degrees of freedom
Residual deviance: 4460.2 on 5993 degrees of freedom
AIC: 4464

Number of Fisher Scoring iterations: 6
Ex. 12.3 (cont.). We select top 5 differently expressed by using two-sample $t$-test and fit logistic regression along with other variables.

```r
> autism = read.csv("autism.csv")  #reading the data
> aut.glm = glm(Autism ~ . , family=binomial, data=autism)
> #fitting the model
> summary(aut.glm)  #summarize the fit
```

Call:
```
glm(formula = Autism ~ ., family = binomial, data = autism)
```

Deviance Residuals:
Min  1Q  Median  3Q  Max
-2.4105  -0.5834  -0.1647  0.4863  2.5613

Coefficients:

|             | Estimate | Std. Error | z value | Pr(>|z|) |
|--------------|----------|------------|---------|----------|
| (Intercept)  | -1.33425 | 2.56463 | -0.520  | 0.602889 |
| GenderM      | 0.14585  | 0.73279  | 0.199   | 0.842233 |
| Age          | -0.05945 | 0.02871  | -2.071  | 0.038365 *|
| SiteM        | -3.43602 | 0.95416  | -3.601  | 0.000317 ***|
| Reg          | 1.17445  | 0.57933  | 2.027   | 0.042636 *|
| Gene1        | -0.10237 | 0.14148  | -0.724  | 0.469332 |
| Gene2        | 0.43250  | 0.32752  | 1.321   | 0.186658 |
| Gene3        | 0.78675  | 0.26275  | 2.994   | 0.002751 **|
| Gene5        | -0.66137 | 0.30426  | -2.174  | 0.029729 *|
| NA.          | 0.08676  | 0.26373  | 0.329   | 0.742165 |

---

Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 143.212  on 103  degrees of freedom
Residual deviance: 74.617  on  94  degrees of freedom
AIC: 94.617
We now select model by using stepwise procedure \texttt{step(aut.glm)}. It selects the model:

\begin{verbatim}
> aut.glm1 = glm(Autism ~ Age + Site + Reg + Gene3 + Gene5,
>                 family=binomial, data=autism)
> summary(aut.glm1)  #summarize the fit
\end{verbatim}

\begin{verbatim}
          Estimate  Std. Error      z value  Pr(>|z|)
(Intercept)  0.01125   2.12388     0.005    0.995773
 Age        -0.06377   0.02804    -2.275     0.022928 *
 SiteM       -3.31923   0.85777    -3.870    0.000109 ***
Reg          1.05110   0.52212     2.013     0.044099 *
Gene3        0.89643   0.22623     3.962   7.42e-05 ***
Gene5       -0.51391   0.18172    -2.828    0.004684 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
\end{verbatim}

12.3 Classification

\textbf{Classification}: Given $X = x$, classify it as “class 1” if
\[ P(Y = 1|X = x) = F(\hat{\beta}_0 + x^T\hat{\beta}) > 0.5 \]

**Example 12.5** Logistic regression and linear decision boundary

Classifier is the same as \( I(\hat{\beta}_0 + x^T\hat{\beta} > 0) \).

**Misclassification rate:** Given \( n^* \) test data, it is defined by

\[
\frac{1}{n^*} \sum_{i=1}^{n^*} I(\hat{y}_i \neq y_i^*) \quad \text{— Hamming distance}
\]

**Ex. 12.3** (cont.). We now use the second model to classify.

```r
> logit = predict(aut.glm1)  # fitted log(odd-ratios)
> prob = exp(logit)/(1+exp(logit))  # fitted probability
> classification = (prob > 0.5)  # classification

### equivalent to directly using (logit > 0)
```
> mean(autism[,1] != classification)  #compute misclassification rate
[1] 0.1346154

**Bayes classifier:** \( f_B(x) = \arg\max_y P(Y = y|X = x) \).

**Likelihood and prior:** By Bayes formula,

\[
P(Y = y|X = x) = \frac{P(X = x|Y = y) p(Y = y)}{P(X = x)}
\]

Optimal choice depends only on the numerator.

**Risk:** For a classifier \( f(X) \), its risk is \( R(f) = P(f(X) \neq Y) \)

**Example** 12.6 *Fisher Discriminants: Normal populations*

Assume \( P(Y = 0) = 0.5 \) and for population \( y = 0 \) or 1,

\[
P(X = x|Y = y) = \frac{1}{\sqrt{(2\pi)^d|\Sigma_y|}} \exp\left(-\frac{1}{2}(x - \mu_y)^T \Sigma_y^{-1}(x - \mu_y)\right).
\]
Bayes risk
\[ R(f^*) \]

Bayes risk

Figure 12.4: Illustration of Bayes classifier and its associated risk when \( P(Y = 0) = 0.5 \). It compares the likelihood ratio. Green data is classified as green on the left.

Then Bayes rule is the log-likelihood ratio: class 0 if

\[
(x - \mu_0)^T \Sigma_0^{-1} (x - \mu_0) + \log |\Sigma_0| \leq (x - \mu_1)^T \Sigma_1^{-1} (x - \mu_1) + \log |\Sigma_1|.
\]

This is a **nearest centroid** classifier.

When \( \Sigma_0 = \Sigma_1 \), it becomes

\[
(\mu_1 - \mu_0)^T \Sigma^{-1} (x - (\mu_0 + \mu_1)/2) < 0.
\]
Figure 12.5: Illustration of decision boundary by nearest centroid classifier. Case 1 ($\Sigma_0 = \Sigma_1$): a linear decision boundary. Case 2 ($\Sigma_0 \neq \Sigma_1$): Quadratic decision boundary.

### 12.4 Support Vector Machine

**Relabel** $y$ as $\pm 1$. Let $f(x)$ be a classifier with decision $\text{sgn}(f(x))$.

**Zero-one Loss**: $L(f(x), y) = I(y \ast f(x) \leq 0)$

**Logit regression**: $\log \frac{p}{q} = f(x)$. Then,

$$L(f(x), y) = \log(1 + e^{-y*f(x)})$$
Support Vector Machine:
Use hinge loss $L(f, y) = (1 - y \cdot f)^+$

Estimation: Find $\beta_0$ and $\beta$ to minimize the empirical loss

$$\sum_i L(f(x_i), y_i) = \sum_i L(\beta_0 + x_i^T \beta, y_i)$$

12.5 Clustering

k-means algorithm: Find clusters $\{C_j\}$ and centroids $\{c_j\}$ to min

$$\sum_{j=1}^k \sum_{i \in C_j} \|x_i - c_j\|^2$$
★ Given $k$ cluster centers $\{c_j\}_{j=1}^k$, classify each data into $k$ clusters by the nearest centroid.

★ Given $k$ clusters, update cluster centers by taking their averages.

★ iterate until convergence.

> library(datasets)  # get the data set "iris"
> iris[1:3,]  # first 3 cases of the data
   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1    5.1      3.5       1.4       0.2  setosa
2    4.9      3.0       1.4       0.2  setosa
3    4.7      3.2       1.3       0.2  setosa
> irisCluster <- kmeans(iris[, 3:4], 3, nstart = 20)
  # 20 random initial choices of centroids, using variables 3 and 4
> irisCluster # show the results
K-means clustering with 3 clusters of sizes 50, 52, 48

Cluster means:

<table>
<thead>
<tr>
<th></th>
<th>Petal.Length</th>
<th>Petal.Width</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.462000</td>
<td>0.246000</td>
</tr>
<tr>
<td>2</td>
<td>4.269231</td>
<td>1.342308</td>
</tr>
<tr>
<td>3</td>
<td>5.595833</td>
<td>2.037500</td>
</tr>
</tbody>
</table>

Clustering vector:

[1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

Within cluster sum of squares by cluster:


(between_SS / total_SS = 94.3 %)

Available components:

[1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss" "size"
[8] "iter" "ifault"

> table(irisCluster$cluster, iris$Species)

<table>
<thead>
<tr>
<th></th>
<th>setosa</th>
<th>versicolor</th>
<th>virginica</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>50</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>48</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>2</td>
<td>46</td>
</tr>
</tbody>
</table>
Hierarchical Clustering:

★ Initially, each object is assigned to its own cluster.

★ at each stage joining the two most similar clusters, continuing until there is just a single cluster.

★ at each stage distances between clusters are recomputed by the LanceWilliams dissimilarity update formula.

> clusters <- hclust(dist(iris[, 3:4]))
> plot(clusters,col="blue")
Cluster Dendrogram

dist(iris[, 3:4])
hclust (", "complete")

Figure 12.6: Hierarchical clustering by using iris data.