# Day 5: Classification

ME314: Introduction to Data Science and Machine Learning

LSE Methods Summer Programme 2021

28 June 2021

# Day 5 Outline

Classification

- Logistic Regression
- Maximum Likelihood
- Multiple logistic regression
- Logistic regression with more than two classes
- Naive Bayes Classifier

Characterizing performance of classifiers

- Confusion matrix
- Sensitivity and specificity
- Performance measures for classifiers: Zoo

# Classification

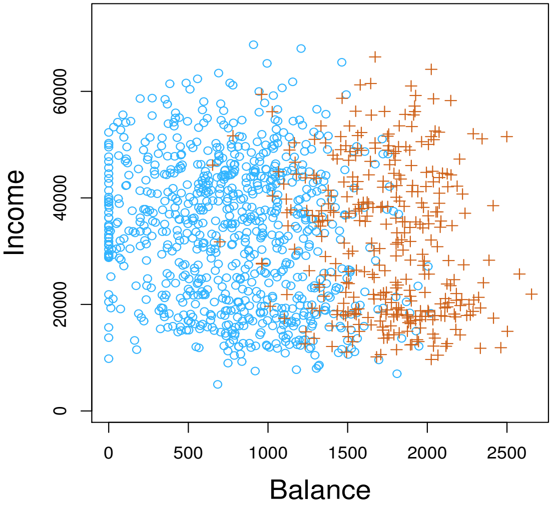
# Classification

- Qualitative variables take values in an unordered set $\mathcal{C}$, such as: *eye color* $\in$ {*brown,blue,green*}; *email* $\in$ {*spam, ham*}.
- Given a feature vector $X$ and a qualitative response $Y$ taking values in the set $\mathcal{C}$, the classification task is to build a function $\mathcal{C}(\mathcal{X})$ that takes as input the feature vector $X$ and predicts its value for $Y$; i.e. $\mathcal{C}(\mathcal{X}) \in \mathcal{C}$.

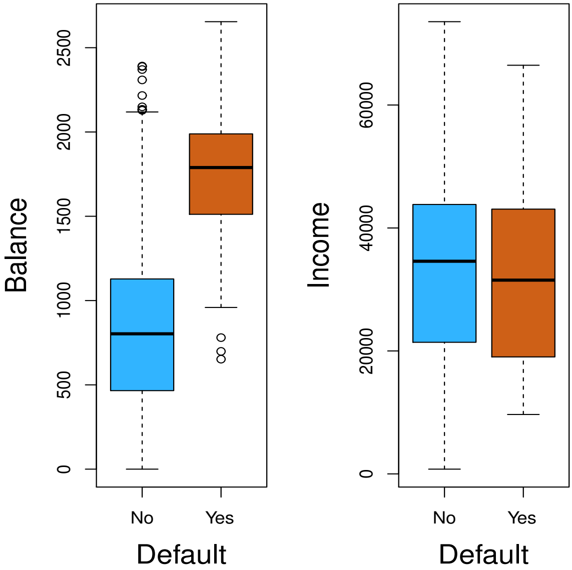
# Classification

- Often we are more interested in estimating the probabilities that $X$ belongs to each category in $\mathcal{C}$.
- For example, it is more valuable to have an estimate of the probability that an insurance claim is fraudulent, than a classification fraudulent or not.

# Example: Credit Card Default



# Example: Credit Card Default



# Can we use Linear Regression?

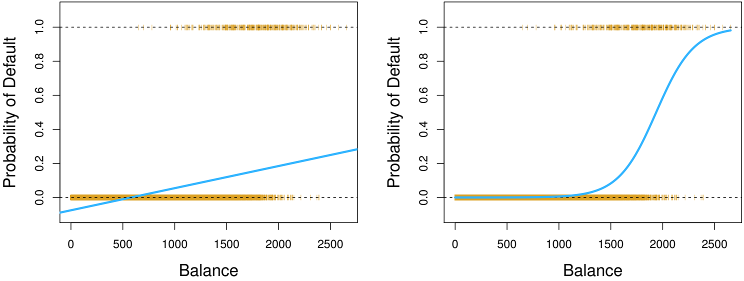
Suppose for the Default classification task that we code

$$Y = \begin{cases} 0 & \text{if } \mathit{No} \\ 1 & \text{if } \mathit{Yes}. \end{cases}$$

Can we simply perform a linear regression of Y on X and classify as Yes if $\widehat{Y} > 0.5$?

- In this case of a binary outcome, linear regression does a good job as a classifier, and is equivalent to linear discriminant analysis which we discuss later.
- Since in the population $E(Y|X = x) = Pr(Y = 1|X = x)$, we might think that regression is perfect for this task.
- However, linear regression might produce probabilities less than zero or bigger than one. Logistic regression is more appropriate.

# Linear versus Logistic Regression



- The orange marks indicate the response $Y$, either 0 or 1.
- Linear regression does not estimate $Pr(Y = 1|X)$ well.
- Logistic regression seems well suited to the task.

# Linear Regression continued

- Now suppose we have a response variable with three possible values. A patient presents at the emergency room, and we must classify them according to their symptoms.

$$Y = \begin{cases} 1 & \text{if } \textit{stroke}; \\ 2 & \text{if } \textit{drug overdose}; \\ 3 & \text{if } \textit{epileptic seizure}. \end{cases}$$

- This coding suggests an ordering, and in fact implies that the difference between *stroke* and *drug overdose* is the same as between *drug overdose* and *epileptic seizure*.
- Linear regression is not appropriate here.
- Multiclass Logistic Regression or Discriminant Analysis are more appropriate.

# Logistic Regression

- Let's write $p(X) = Pr(Y = 1|X)$ for short and consider using *balance* to predict *default*. Logistic regression uses the form

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}.$$

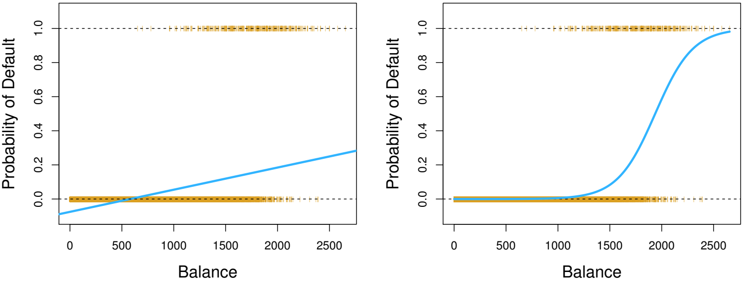
  ($e \approx 2.71828$ is a mathematical constant [ Euler's number.])

- It is easy to see that no matter what values $\beta_0$, $\beta_1$ or $X$ take, $p(X)$ will have values between 0 and 1.
- A bit of rearrangement gives

$$log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X.$$

- This monotone transformation is called the log odds or logit transformation of $p(X)$.

# Linear versus Logistic Regression



- Logistic regression ensures that our estimate for $p(X)$ lies between 0 and 1.

# Maximum Likelihood

- We use maximum likelihood to estimate the parameters.

$$\ell(\beta_0, \beta) = \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1 - p(x_i)).$$

- This likelihood gives the probability of the observed zeros and ones in the data.
- We pick $\beta_0$ and $\beta_1$ to maximize the likelihood of the observed data.
- Most statistical packages can fit linear logistic regression models by maximum likelihood. In R we use the *glm* function.

```r
library(ISLR)
data("Default")
names(Default)
```

```
## [1] "default" "student" "balance" "income"
```

```r
logistic <- glm(Default$default ~ Default$balance, family = binomial)
```

```
summary(logistic)

##
## Call:
## glm(formula = Default$default ~ Default$balance, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2697  -0.1465  -0.0589  -0.0221   3.7589
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -1.065e+01  3.612e-01  -29.49   <2e-16 ***
## Default$balance  5.499e-03  2.204e-04   24.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1596.5  on 9998  degrees of freedom
## AIC: 1600.5
##
## Number of Fisher Scoring iterations: 8
```

# Making Predictions

- What is our estimated probability of *default* for someone with a balance of \$1000?

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} = \frac{e^{-10.6513 + 0.0055 \times 1000}}{1 + e^{-10.6513 + 0.0055 \times 1000}} = 0.006$$

- With a balance of \$2000?

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} = \frac{e^{-10.6513 + 0.0055 \times 2000}}{1 + e^{-10.6513 + 0.0055 \times 2000}} = 0.586$$

```
logistic2 <- glm(Default$default ~ Default$student, family = binomial)
summary(logistic2)

##
## Call:
## glm(formula = Default$default ~ Default$student, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2970  -0.2970  -0.2434  -0.2434   2.6585
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -3.50413    0.07071  -49.55  < 2e-16 ***
## Default$studentYes  0.40489    0.11502    3.52 0.000431 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 2908.7  on 9998  degrees of freedom
## AIC: 2912.7
##
## Number of Fisher Scoring iterations: 6
```

# Making Predictions (binary variable)

$$\widehat{Pr}(default = Yes|student = Yes) = \frac{e^{-3.5041+0.4049\times 1}}{1 + e^{-3.5041+0.4049\times 1}} = 0.0431$$

$$\widehat{Pr}(default = Yes|student = No) = \frac{e^{-3.5041+0.4049\times 0}}{1 + e^{-3.5041+0.4049\times 0}} = 0.0292$$

# Logistic regression with several variables

$$log\left(\frac{p(X)}{1-p(X)}\right)=\beta_0+\beta_1 X_1+\ldots+\beta_p X_p$$

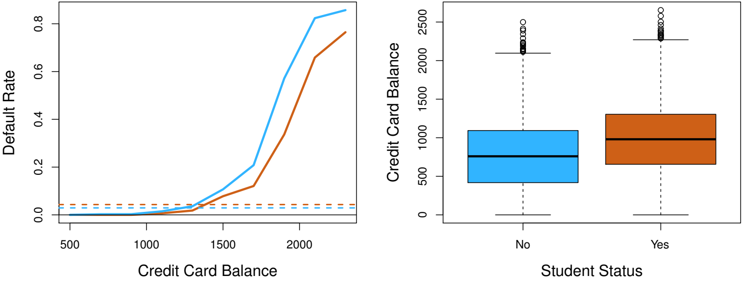
$$p(X)=\frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}$$

```r
logistic3 <- glm(Default$default ~ Default$balance + Default$income + Default$student, family = binomial)
summary(logistic3)
```

```
##
## Call:
## glm(formula = Default$default ~ Default$balance + Default$income +
##     Default$student, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4691  -0.1418  -0.0557  -0.0203   3.7383
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
## Default$balance       5.737e-03  2.319e-04  24.738  < 2e-16 ***
## Default$income        3.033e-06  8.203e-06   0.370  0.71152
## Default$studentYes   -6.468e-01  2.363e-01  -2.738  0.00619 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1571.5  on 9996  degrees of freedom
## AIC: 1579.5
##
## Number of Fisher Scoring iterations: 8
```

- Why is coefficient for *student* negative, while it was positive before?

# Confounding



- Students tend to have higher balances than non-students, so their marginal default rate is higher than for non-students.
- But for each level of balance, students default less than non-students.
- Multiple logistic regression can tease this out.

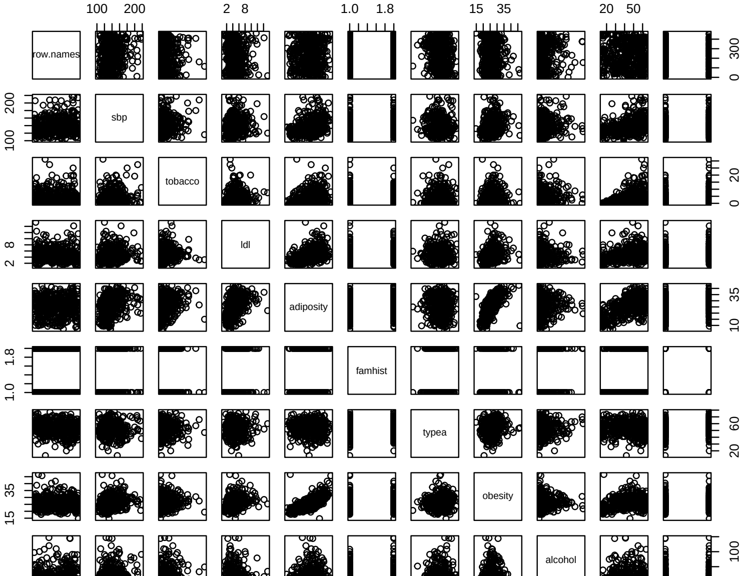
# Example: South African Heart Disease

- 160 cases of MI (myocardial infarction) and 302 controls (all male in age range 15-64), from Western Cape, South Africa in early 80s.
- Overall prevalence very high in this region: 5.1%.
- Measurements on seven predictors (risk factors), shown in scatterplot matrix.
- Goal is to identify relative strengths and directions of risk factors.
- This was part of an intervention study aimed at educating the public on healthier diets.

```
SAheart <- read.csv("http://statweb.stanford.edu/~tibs/ElemStatL
                     row.names = NULL)
dplyr::glimpse(SAheart)
```

```
## Rows: 462
## Columns: 11
## $ row.names <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13,
## $ sbp       <int> 160, 144, 118, 170, 134, 132, 142, 114, 114
## $ tobacco   <dbl> 12.00, 0.01, 0.08, 7.50, 13.60, 6.20, 4.05,
## $ ldl       <dbl> 5.73, 4.41, 3.48, 6.41, 3.50, 6.47, 3.38, 4
## $ adiposity <dbl> 23.11, 28.61, 32.28, 38.03, 27.78, 36.21, 1
## $ famhist   <chr> "Present", "Absent", "Present", "Present",
## $ typea     <int> 49, 55, 52, 51, 60, 62, 59, 62, 49, 69, 72,
## $ obesity   <dbl> 25.30, 28.87, 29.14, 31.99, 25.99, 30.77, 2
## $ alcohol   <dbl> 97.20, 2.06, 3.81, 24.26, 57.34, 14.14, 2.6
## $ age       <int> 52, 63, 46, 58, 49, 45, 38, 58, 29, 53, 60,
## $ chd       <int> 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0
```

```r
SAheart$famhist <- factor(SAheart$famhist)
pairs(SAheart)
```



```
heartfit <- glm(chd ~ . , data = SAheart, family = binomial)
summary(heartfit)

##
## Call:
## glm(formula = chd ~ ., family = binomial, data = SAheart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8320  -0.8250  -0.4354   0.8747   2.5503
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -5.9207616  1.3265724  -4.463 8.07e-06 ***
## row.names      -0.0008844  0.0008950  -0.988 0.323042
## sbp             0.0076602  0.0058574   1.308 0.190942
## tobacco         0.0777962  0.0266602   2.918 0.003522 **
## ldl             0.1701708  0.0597998   2.846 0.004432 **
## adiposity       0.0209609  0.0294496   0.712 0.476617
## famhistPresent  0.9385467  0.2287202   4.103 4.07e-05 ***
## typea           0.0376529  0.0124706   3.019 0.002533 **
## obesity        -0.0661926  0.0443180  -1.494 0.135285
## alcohol         0.0004222  0.0045053   0.094 0.925346
## age             0.0441808  0.0121784   3.628 0.000286 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 596.11  on 461  degrees of freedom
## Residual deviance: 471.16  on 451  degrees of freedom
## AIC: 493.16
##
## Number of Fisher Scoring iterations: 5
```

# Logistic regression with more than two classes

- So far we have discussed logistic regression with two classes.
- It is easily generalized to more than two classes.
- One version (used in the R package *glmnet*) has the symmetric form

$$Pr(Y = k|X) = \frac{e^{\beta_{0k}+\beta_{1k}X_1+\ldots+\beta_{pk}X_p}}{\sum_{\ell=1}^{K} e^{\beta_{0\ell}+\beta_{1\ell}X_1+\ldots+\beta_{p\ell}X_p}}$$

- Here there is a linear function for each class.
- Multiclass logistic regression is also referred to as multinomial regression.

# Naive Bayes Classifier

- Naive Bayes (NB) classifier especially appropriate when the dimension $p$ of the feature space is high, making density estimation unattractive.
- Assumes that given a class $G = j$, the features $X_k$ are independent:

$$f_j(X) = \prod_{k=1}^{p} f_{jk}(X_k).$$

- While this assumption is pretty heroic and generally not true, it significantly simplifies the estimation.
- The individual class-conditional marginal densities $f_{jk}$ can each be estimated separately.
- If a component $X_j$ of $X$ is discrete, then an appropriate histogram estimate can be used. This provides a seamless way of mixing variable types in a feature vector.

# Naive Bayes Classifier

- Despite these strong assumptions, NB classifiers often outperform far more sophisticated alternatives.
- Although the individual class density estimates may be biased, this bias might not hurt the posterior probabilities as much, especially near the decision regions.
- In fact, the problem may be able to withstand considerable bias for the savings in variance such a “naive” assumption earns.

# Characterizing performance of classifiers

# Confusion matrix and error rates (from LDA)

| | | True | Default | Status |
|---|---|---|---|---|
| | | No | Yes | Total |
| Predicted | No | 9644 | 252 | 9896 |
| Default Status | Yes | 23 | 81 | 104 |
| | Total | 9667 | 333 | 10000 |

- (23 + 252) / 10000 errors — a 2.75% misclassification rate.
- Some caveats:
    - This is training error, and we may be overfitting. Not a big concern here since $n = 10000$ and $p = 4$.
    - If we classified to the prior – always to class *No* in this case – we would make 333/10000 errors, or only 3.33%.
    - Of the true *No*'s, we make 23/9667 = 0.2% errors; of the true *Yes*'s, we make 252/333 = 75.7% errors!

# Types of errors

- False positive rate: The fraction of negative examples that are classified as positive – 0.2% in example.
- False negative rate: The fraction of positive examples that are classified as negative – 75.7% in example.

# Sensitivity and specificity

- Performance of a classifier is often characterized in terms of sensitivity and specificity.
- Here, the sensitivity is the percentage of true defaulters that are identified. It is 24.3% in our case.
- The specificity is the percentage of non-defaulters that are correctly identified. Here it is $(1 - 23/9,667) \cdot 100 = 99.8\%$
- The true positive rate is the sensitivity of our classifier.
- The false positive rate is *one minus* the specificity of our classifier.

# Errors and threshold

- We produced the confusion matrix above by classifying to class *Yes* if

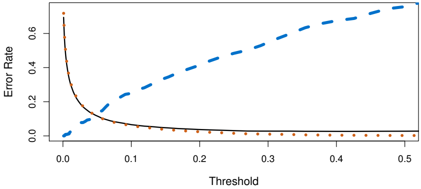
$$\widehat{Pr}(Default = Yes|Balance, Student) \geq 0.5$$

- We can change the two error rates by changing the threshold from 0.5 to some other value in [0,1]:

$$\widehat{Pr}(Default = Yes|Balance, Student) \geq threshold,$$

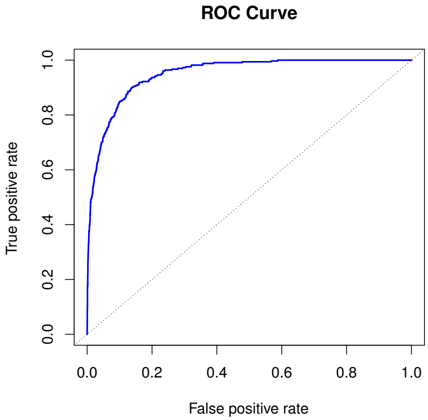
and vary *threshold*.

# Varying the *threshold*



- Error rates are shown as a function of the threshold value for the posterior probability that is used to perform the assignment.
- The black solid line displays the overall error rate.
- The blue dashed line represents the fraction of defaulting customers that are incorrectly classified (False Negative).
- The orange dotted line indicates the fraction of errors among the non-defaulting customers (False Positive).
- In order to reduce the false negative rate, we may want to reduce the threshold to 0.1 or less.

# ROC curve



- The ROC plot displays both simultaneously.
- Sometimes we use the AUC or area under the curve to summarize the overall performance.
- Higher AUC is good.

# Characterizing performance of classifiers

| | | Predicted | class | |
|---|---|---|---|---|
| | | - or Null | + or Non-null | Total |
| True | - or Null | True Neg. (TN) | False Pos.(FP) | N |
| class | + or Non-null | False Neg. (FN) | True Pos. (TP) | P |
| | Total | N* | P* | |

- “+” is “disease” or alternative (non-null) hypothesis (here, those who default);
- “-” is “non-disease” or the null hypothesis (here, those who do not default).

# Performance measures for classifiers

| Name | Definition | Synonyms |
|---|---|---|
| False Pos. rate | FP/N | Type I error, 1- Specificity |
| True Pos. rate | TP/P | 1 - Type II error, power, sensitivity, recall |
| Pos. Pred. value | TP/P* | Precision, 1-false discovery proportion |
| Neg. Pred. value | TN/N* | |

- The denominators for the false positive and true positive rates are the actual population counts in each class.
- The denominators for the positive predictive value and the negative predictive value are the total predicted counts for each class.

# Summary

- Logistic regression is very popular for classification, especially when $K = 2$.
- LDA is useful when $n$ is small, or the classes are well separated, and Gaussian assumptions are reasonable. Also when $K > 2$.
- Naive Bayes is useful when $p$ is very large.
- See Section 4.5 for some comparisons of logistic regression, LDA and KNN.