Methods of classification

- introduction
- loss function / measurement of how good a classifier is
- logistic regression
- linear regression
- discriminant analysis (LDA, QDA)
- KNN
- support vector machine
- trees
- neural networks

} → classification + regression

$y_i \in \{0; 1\}$, $y_i \in \{0, \dots, K\}$

classes

Idea: allocate a statical unit to a class (category) based on the information of some variables

$K > 2$ is the most frequent case, typical examples

- health / disease — when using a medical test
- solvent / non solvent — when assign / not assign a loan
- customers receptive / not receptive to a marketing campain

GOAL: find a prediction rule $f(x)$ which allocates the new statistical unit to right category

In the book: through "juice example"
- $n = 1070$ (purchases on stores) $\quad y_i \in \{CH, MM\}$ two brands of juice
- information about
  - price (price CH and price MM)
  - discount applied (discount CH and discount MM)
  - identifier for the week
  - " " " store (5 stores)
  - loyalty to the brand (% of time a subject purchases the juice of brand MM, loyalty MM)

DATA SPLIT → 75% of data for training
→ 25% " " " testing

We already saw the logistic regression
π = probability of buying MM

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 \text{week} + \beta_2 \text{price CH} + \beta_3 \text{price MM} + \dots + \beta_7^T I_{\text{store}}$$

$\beta_7 = (\beta_{7A}, \beta_{7B}, \beta_{7C}, \beta_{7D})$

$I_{\text{store}} =$ (2nd store, 3rd store)
1 → (0 0 …)
2 → (1 0 …)
3 → (0 1 …)
4 → (0 0 …)
5 → (0 0 …)

$\hat{\pi}$ Assign a new observation to {MM, CH}

$\text{logit}\left(\frac{\hat{\pi}}{1-\hat{\pi}}\right) = \hat{\eta}$ $\quad \hat{\pi} = \frac{e^{\hat{\eta}}}{1+e^{\hat{\eta}}}$ $\quad \hat{\pi} > 0.5 \Rightarrow$ MM, $\hat{\pi} < 0.5 \Rightarrow$ CH

→ confusion matrix

prediction (by the model) $\hat{y}_{new}$		actual value (y_{new}) CH	MM	total
	CH	150	23	173
	MM	19	76	95
total		169	99	268 → n_t

misclassification error $\frac{19+23}{268} = 0.157$

In general

prediction	true value −	true value +
−	true negative	false negative
+	false positive	true positive

type I error → α = false positive

false positive = reject H_0 when H_0 true → $\frac{\text{false positive}}{\text{total positive}} = P(\text{false positive})$

false negative = not reject H_0 when H_0 not true → $\frac{\text{false neg.}}{\text{tot. negative}} = P(\text{false negative})$

type II error → β

Note: using misclassification error, we are giving the same weight to both kinds of error. There are cases when this is not correct so we may need to weight differently the errors → different cost

prediction	true pizza	true whatever pizza with pineapple on that
pizza	0	10 / 120
w. pineapple	1	0

cost

move $\hat{\pi}^*$ from 0.5

What happens when moving the threshold $\hat{y}^*$ from the default 0.5

prediction \ true value	−	+
−	↑ ↓	↑ ↓
+	↓ ↑	↓ ↑

for increasing $\hat{y}^*$:
- less false positive
- less true positive
- more false negative
- more true negative → higher specificity

proportion of predicted negative over the true negative $1-\alpha$

for decreasing $\hat{y}^*$:
- less false negative
- less true negative
- more false positive
- more true positive → higher sensitivity

proportion of predicted positive over true positive $1-\beta$

Our goal is to have the highest specificity and the highest sensitivity as possible. Unfortunately, increasing one usually means decreasing the other

To evaluate sensitivity and specificity and their changes when moving $\hat{y}^*$, we plot (1 − specificity) versus sensitivity

↳ ROC curve (Receiver Operating Characteristic)
- we want the curve to be as 'top left' as possible
- intercept = random choice

Lift curve

- provides a measure of the improvement of classification with a model w.r.t. classification by chance

E.g.: one is interested in sending advertisement to customer which buy MM. It costs money, so it is possible to send only N letters
- random choice: select N people randomly
- model-based: select the N people with highest $\hat{\pi}$

prediction \ true value	−	+	
−	n_{00}	n_{01}	$n_{0\cdot}$
+	n_{10}	n_{11}	N
	$n_{\cdot 0}$	$n_{\cdot 1}$	n

$$lift = \frac{n_{11}/N}{n_{\cdot 1}/n}$$

? est. of the
it is the expected improvement w.r.t. random choice

in the example $lift = \frac{76/95}{93/268} \approx 2.17$

Extension to several categories

We have $K > 2$ classes, we need to compute

$$Pr[Y = k \mid X = x] = \hat{\pi}_k(x)$$

Obviously $\sum_{k=0}^{K-1} \hat{\pi}_k(x) = 1$

Similarly to logistic regression

$$\log \frac{\hat{\pi}_k(x)}{\hat{\pi}_0(x)} = \eta_k(x) = \beta_0 + X\beta$$

So $\frac{\hat{\pi}_k(x)}{\hat{\pi}_0(x)} = e^{\eta_k(x)}$ for $k = 1, \dots, K-1$ $\rightarrow \frac{\sum_{k=1}^{K-1} \hat{\pi}_k(x)}{\hat{\pi}_0(x)} = \sum_{k=1}^{K-1} e^{\eta_k(x)}$

Adding 1 on both sides

$$1 + \frac{\sum_{k=1}^{K-1} \hat{\pi}_k(x)}{\hat{\pi}_0(x)} = 1 + \sum_{k=1}^{K-1} e^{\eta_k(x)}$$

$$\frac{\hat{\pi}_0 + \sum_{k=1}^{K-1} \hat{\pi}_k(x)}{\hat{\pi}_0(x)} = 1 + \sum_{k=1}^{K-1} e^{\eta_k(x)} \rightarrow \hat{\pi}_0(x) = \frac{1}{1 + \sum_{k=1}^{K-1} e^{\eta_k(x)}}$$

Since

$$\frac{\hat{\pi}_k(x)}{\hat{\pi}_0(x)} = e^{\eta_k(x)}$$

$$\hat{\pi}_k \left(1 + \sum_{k=1}^{K-1} e^{\eta_k(x)} \right) = e^{\eta_k(x)} \rightarrow \hat{\pi}_k(x) = \frac{e^{\eta_k(x)}}{1 + \sum_{k=1}^{K-1} e^{\eta_k(x)}}$$

This is called multivariate logistic model, and $\eta_k(x)$ are estimated by fitting $K-1$ logistic models, each of these models are comparing class k with the baseline class 0, conditional to y belongs to one of them

$$\log \frac{Pr[Y=k \mid Y=0 \cup Y=k]}{Pr[Y=0 \mid Y=0 \cup Y=k]} = \log \frac{\frac{Pr[Y=k \cap (Y=0 \cup Y=k)]}{Pr[Y=0 \cup Y=k)}}{\frac{Pr[Y=0 \cap (Y=0 \cup Y=k)]}{Pr[Y=0 \cup Y=k)}}$$

$$= \log \frac{\frac{\hat{\pi}_k(x)}{\hat{\pi}_k(x) + \hat{\pi}_0(x)}}{\frac{\hat{\pi}_0(x)}{\hat{\pi}_k(x) + \hat{\pi}_0(x)}}$$

$$= \log \frac{\hat{\pi}_k(x)}{\hat{\pi}_0(x)} = \eta_k(x)$$

The choice of class 0 as baseline is arbitrary, but the desired probabilities do not change

Alternative: multinomial logit model (multinomial regression)

- assume that $\hat{\pi}_0(x), \dots, \hat{\pi}_{K-1}(x)$ are the parameters of a multinomial distribution (allocate n balls in K boxes)

$$f(y_0, \dots, y_{K-1}; \hat{\pi}_0(x), \dots, \hat{\pi}_{K-1}(x)) =$$

$$= Pr[Y_0 = y_0 \cap \dots \cap Y_{K-1} = y_{K-1}]$$

$$= \frac{n!}{y_0! \dots y_{K-1}!} \hat{\pi}_0(x)^{y_0} \times \dots \times \hat{\pi}_{K-1}(x)^{y_{K-1}}$$

- we estimates $\hat{\alpha}_0(x), \dots, \hat{\alpha}_{K-1}(x)$ by maximizing the log-likelihood

$$\ell(\hat{\pi}) = \sum_{k=0}^{K-1} y_k \log \hat{\pi}_k(x)$$

to $\hat{\alpha}_0, \dots, \hat{\alpha}_{K-1}$