UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
School of Professional & Executive Development

Algorithms for linear classification

Lluís A. Belanche
Computer Science Department
belanche@cs.upc.edu

Big Data Management and Analytics

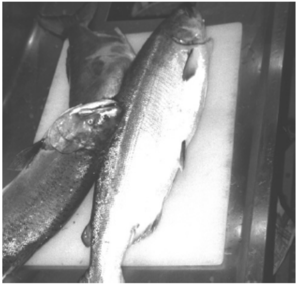
February 22, 2017

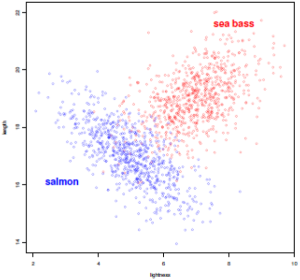
BAYESIAN GENERATIVE CLASSIFIERS

Introduction to classification: an example

Example 1: Fish classification

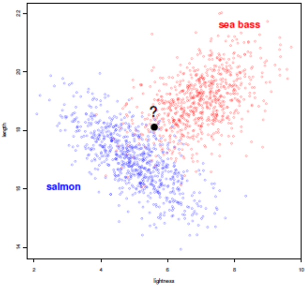
- A fish processing plant wants to automate the process of sorting incoming fish according to species (salmon or sea bass)
- The system consists of a conveyor belt, a robotic arm, a vision system with an overhead CCD camera and a computer
- After some preprocessing, each fish is characterized by two features: average lightness and length





Introduction to classification: an example

Given labeled training data coming from some unknown joint probability distribution, should we predict the new point as salmon or sea bass?



The **goal** is to obtain a model based on training data (*known* examples) with high classification accuracy on future *unknown* examples

$\longrightarrow$ good **generalization**

Introduction: Bayes' formula

Thomas Bayes: XVIII-century priest. His works on the celebrated formula were found upon his death

Discrete random variables

Let A be a discrete r.v. with pmf P_A. We use the shorthand notation $P(a)$ to mean $P_A(A = a)$. Similarly we write $P(b|a)$ to mean $P_{B|A}(B = b|A = a)$, etc, where

$$P(b|a) = \frac{P(b, a)}{P(a)}, \ P(a) > 0$$

(**prior**, **joint** and **conditional** probabilities)

Discrete random variables

Let $\{a_1, \dots, a_n\}, \{b_1, \dots, b_m\}$ the sets of possible values that A, B can take. Then, for any $a \in \{a_1, \dots, a_n\}$:

$$P(a) = \sum_{j=1}^{m} P(a, b_j) = \sum_{j=1}^{m} P(a|b_j)P(b_j)$$

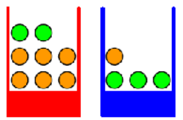
Since $P(a, b) = P(b, a)$, it follows that, for any a_k, b_j:

$$P(b_j|a_k) = \frac{P(a_k|b_j)P(b_j)}{\sum_{i=1}^{m} P(a_k|b_i)P(b_i)}, \qquad \text{with} \sum_{j=1}^{m} P(b_j|a_k) = 1$$

(**posterior** probabilities)

Example

The red box contains 6 oranges and 2 apples, the blue box contains 1 orange and 3 apples. Suppose we pick the red box 40% of the time and the blue box 60% of the time.



1. What is the overall probability that we pick an apple?
2. Given that we have chosen an orange, what is the probability that the box we chose was the blue one?

(from Bishop, C. *Pattern Recognition and Machine Learning*)

Solution

Let us introduce random variables B for box and F for fruit:

- $B = r$ (for red) and $B = b$ (for blue)
- $F = o$ (for orange) and $F = a$ (for apple)

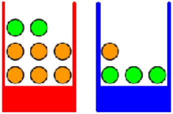
The **prior** probabilities of selecting the red or blue boxes are

$$P(B = r) = \frac{4}{10} \qquad P(B = b) = \frac{6}{10}$$

Solution

Now for the **conditional** probabilities:

$$P(F = a | B = r) = \frac{1}{4}$$
$$P(F = o | B = r) = \frac{3}{4}$$
$$P(F = a | B = b) = \frac{3}{4}$$
$$P(F = o | B = b) = \frac{1}{4}$$



What is the overall (**unconditional**) probability that we pick an apple?

$$\begin{aligned} P(F=a) &= P(F=a|B=r)P(B=r) + P(F=a|B=b)P(B=b) \\ &= \frac{1}{4}\cdot\frac{4}{10} + \frac{3}{4}\cdot\frac{6}{10} = \frac{11}{20} \end{aligned}$$

Therefore $P(F=o) = 1 - \frac{11}{20} = \frac{9}{20}$.

Although there are more oranges in total, picking an apple is more likely!

Solution

Given that we have chosen an orange, what is the **posterior** probability that the box we chose was the blue one?

$$P(B=b|F=o) = \frac{P(F=o|B=b)P(B=b)}{P(F=o)} = \frac{1}{4} \cdot \frac{6}{10} \cdot \frac{20}{9} = \frac{1}{3}$$

$$P(B=r|F=o) = \frac{P(F=o|B=r)P(B=r)}{P(F=o)} = \frac{3}{4} \cdot \frac{4}{10} \cdot \frac{20}{9} = \frac{2}{3}$$

Note that $P(B=b|F=o) + P(B=r|F=o) = 1$, as they should, because conditional distributions are distributions.

Mixed random variables Suppose X is a continuous r.v. and Y is a discrete r.v. with values in $\{y_1, \dots, y_m\}$.
In this case, $p(\cdot|y_i)$ is a continuous r.v. and $P(\cdot|x)$ is a discrete r.v. Moreover,

$$P(y_j|x) = \frac{p(x|y_j)P(y_j)}{\sum_{i=1}^{m} p(x|y_i)P(y_i)}, \qquad \text{with} \sum_{j=1}^{m} P(y_j|x) = 1$$

Generative classifiers

Generative classifiers can be obtained from Bayes formula:

$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{\sum_{i=1}^{K} p(\mathbf{x}|\omega_i)P(\omega_i)}$$

expresses the *posterior* probability that an object with measured feature $\mathbf{x}$ belongs to class $P(\omega_i),\ i \in \Omega = \{1, \ldots, K\}$.

- Upon observing a **feature vector x**, the formula converts **prior** probabilities $P(\omega_i)$ into **posterior** probabilities $P(\omega_i|\mathbf{x})$
- The Bayes rule says:
  "the predicted class of $\mathbf{x}$ is $\arg\max_{i=1,\ldots,K} P(\omega_i|\mathbf{x})$"

The sets $\mathcal{R}_k := \{\mathbf{x} / \hat{\omega}(\mathbf{x}) = k\}$ are called **regions** (and depend on the specific classifier)

A continuous d-variate random vector $\mathbf{X} = (X_1, \ldots, X_d)^\top$ is **normally distributed**, written $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, when its joint pdf is:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}$$

where $\boldsymbol{\mu}$ is the *mean vector* and $\Sigma_{d \times d} = (\sigma_{ij}^2)$ is the (real symmetric and p.d.) *covariance matrix*.

- $\mathbb{E}[\mathbf{X}] = \boldsymbol{\mu}$ and $\mathbb{E}[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^\top] = \Sigma$.
- $CoVar[X_i, X_j] = \sigma_{ij}^2$ and $Var[X_i] = \sigma_{ii}^2$

---

if $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, then X_i, X_j are independent $\iff$ $CoVar[X_i, X_j] = 0$

(in general, only the left-to-right implication holds)

Generative classifiers for the Gaussian density (QDA)

For Gaussian classes, $X_{|\Omega=k} \sim \mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k)$, using Bayes rule and the natural log, a **discriminant function** for class ω_k is:

$$g_k(\mathbf{x}) := \ln \left\{ P(\omega_k) p(\mathbf{x}|\omega_k) \right\} =$$

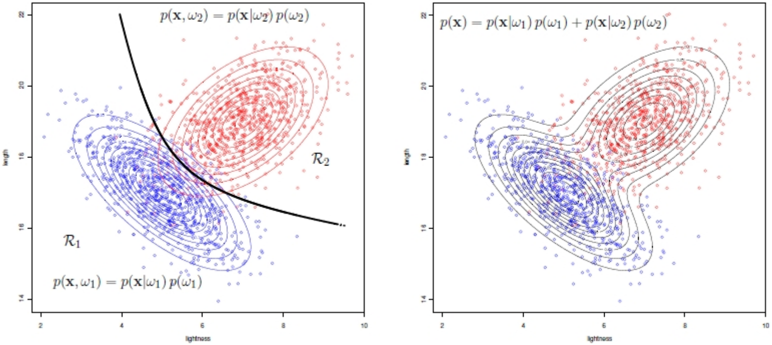
$$\ln\ P(\omega_k) - \ln \left\{ (2\pi)^{\frac{d}{2}} |\Sigma_k|^{\frac{1}{2}} \right\} - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)$$

Eliminating constant terms:

$$g_k(\mathbf{x}) = \ln\ P(\omega_k) - \frac{1}{2} \left(\ln\ |\Sigma_k| + (\mathbf{x} - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right)$$

QDA

This expression is called a **quadratic discriminant function**; the boundaries $g_i(\mathbf{x}) = g_j(\mathbf{x})$ are general hyperquadrics



Solution to the Fish factory example

(from Duda, Hart & Stork *Pattern Classification*, Wiley, 2001)

The Bayes rule says: "if $P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x})$ then ω_1 else ω_2"

If we assume that all class-conditional distributions $p(\mathbf{x}|\omega_k)$ have the **same covariance** matrix Σ, we get:

$$g_k(\mathbf{x}) = \ln\ P(\omega_k) + \boldsymbol{\mu}_k^\top \Sigma^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_k^\top \Sigma^{-1}\boldsymbol{\mu}_k$$

Reorganizing terms we obtain:

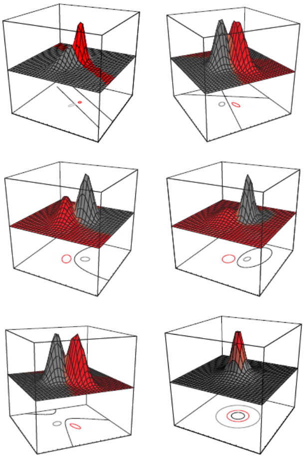
$$g_k(\mathbf{x}) = \ln\ P(\omega_k) + \boldsymbol{\mu}_k^\top \Sigma^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_k^\top \Sigma^{-1}\boldsymbol{\mu}_k = \boldsymbol{\beta}_k^\top\mathbf{x} + \beta_{k0}$$

where $\boldsymbol{\beta}_k = \Sigma^{-1}\boldsymbol{\mu}_k$ and $\beta_{k0} = -\frac{1}{2}\boldsymbol{\mu}_k^\top \Sigma^{-1}\boldsymbol{\mu}_k + \ln\ P(\omega_k)$

LDA

This expression is called a **linear discriminant function**; the boundaries $g_i(\mathbf{x}) = g_j(\mathbf{x})$ are hyperplanes

Generative classifiers for the Gaussian density



(from Duda, Hart & Stork *Pattern Classification*, Wiley, 2001)

Generative classifiers for the Gaussian density

- If we further assume that all the X_i, X_j are **statistically independent**, that is $\Sigma = diag(\sigma_1^2, \ldots, \sigma_d^2)$, we get:

$$g_k(\mathbf{x}) = \ln\ P(\omega_k) - \frac{1}{2}\sum_{i=1}^{d}\frac{(\mu_{ki} - x_i)^2}{\sigma_i^2}$$

- If we further assume that all the X_i have the **same variance** σ^2, that is $\Sigma = \sigma^2 I_d$, we get:

$$g_k(\mathbf{x}) = \ln\ P(\omega_k) - \frac{1}{2\sigma^2}\|\boldsymbol{\mu}_k - \mathbf{x}\|^2$$

- If we further assume that all the classes have the **same prior** $P(\omega_k) = \frac{1}{K}$, we get:

$$g_k(\mathbf{x}) = -\|\boldsymbol{\mu}_k - \mathbf{x}\|^2$$

Computations in practice

In practice, only an i.i.d data sample D is available. Let $D_k \subset D$ be the subset of observations belonging to class ω_k ($D_1, \ldots, D_K$ is a partition of D). We use the unbiased estimates:

$$\hat{\boldsymbol{\mu}}_k = \frac{1}{|D_k|} \sum_{\mathbf{x} \in D_k} \mathbf{x}; \qquad \hat{P}(\omega_k) = \frac{|D_k|}{|D|}$$

1. If we know (or assume) that covariance matrices are different (wish to use QDA):

$$\hat{\Sigma}_k = \frac{1}{|D_k| - 1} \sum_{\mathbf{x} \in D_k} (\mathbf{x} - \hat{\boldsymbol{\mu}}_k)(\mathbf{x} - \hat{\boldsymbol{\mu}}_k)^\top$$

2. If we know (or assume) that covariance matrices are equal (wish to use LDA):

$$\hat{\Sigma}_{\text{pooled}} = \frac{1}{|D| - K} \sum_{k=1}^{K} (|D_k| - 1)\hat{\Sigma}_k$$

Key issues (I)

- The Bayes classifier is the best possible classifier when the class-conditional densities and priors are known
- In all cases, we have a **minimum-distance classifier**:
  - In the general case (some covariance matrices are different), the classifier is called **quadratic discriminant analysis** (QDA)
  - In case all covariance matrices are equal, the classifier is called **linear discriminant analysis** (LDA)
- Therefore using a specific distance function corresponds to certain statistical assumptions
- These methods are well-principled, fast and reliable

Key issues (II)

- LDA can also be used for dimension reduction (it is known as **Fisher's linear discriminant** or FDA)
- The question whether the assumptions hold can rarely be answered in practice; in most cases we are limited to posing and answering the question

  "*does this classifier give satisfactory predictions or not?* "

- If the class-conditional densities are far from the assumptions (e.g. being Gaussian), the model will be poor; even when they are close, sample statistics should be estimated reliably

Regularized Discriminant Analysis

- If the number of variables d is higher than the number of observations of a group $|D_k|$ and lower than the total number of observations N, QDA cannot be applied, because the class covariance matrix $\hat{\Sigma}_k$ is singular
- If the number of variables d is higher than the total number of observations N, neither QDA nor LDA can be used, because both $\hat{\Sigma}_k$ and $\hat{\Sigma}_{\mathrm{pooled}}$ are singular
- These problems can be overcome by applying **regularization**:

$$\hat{\Sigma}_k(\lambda,\gamma) = (1-\gamma)\hat{\Sigma}_k(\lambda) + \frac{\gamma}{d} Tr\left[\hat{\Sigma}_k(\lambda)\right] I_d$$

where $\hat{\Sigma}_k(\lambda) = (1-\lambda)\hat{\Sigma}_k + \lambda\hat{\Sigma}_{\mathrm{pooled}}$

LDA is $(\lambda,\gamma) = (1,0)$ and QDA is $(\lambda,\gamma) = (0,0)$

$$\boxed{\text{DISCRIMINATIVE CLASSIFIERS}}$$

Generalized Linear Models (REMINDER)

GLMs allow for general conditional target distributions:

$$g(\mathbb{E}[T_n|\mathbf{X_n}]) = \boldsymbol{\beta}^\top \mathbf{X}_n + \beta_0$$

Generalized Linear Model:

- A GLM is a linear predictor of a convenient function of the expected value of the target variable, **conditioned** on the predictors
- This convenient function g is typically a smooth invertible function and called the **link function**
- The T_n are taken as i.i.d. and drawn from a distribution of the **exponential family** (Poisson, Gaussian/Normal, Chi-squared, Bernoulli, Gamma, Beta, ...)

The GLM setup then asks for a model $y_k(\mathbf{x})$ such that:

$$y_k(\mathbf{x}) = g^{-1}(\boldsymbol{\beta}_k^\top \mathbf{x} + \beta_{k0})$$

- where g is a convenient "interface" function ...
- ... and try to optimize the $\boldsymbol{\beta}_k$ and β_{k0} parameters directly
- No distributional assumptions on the $\mathbf{x}$!
- We must decide a distribution for the t given the $\mathbf{x}$!
- Yes, but where are the statistical assumptions?

For two classes ($K = 2$), we **model** the posterior probability for class ω_1 as:

$$y(\mathbf{x}) = P(\omega_1|\mathbf{x})$$

The idea is that the distribution is the Bernoulli:

$$T_n|X_n \sim Ber(p_n)$$

and

$$y(\mathbf{x}_n) = p_n = g^{-1}(\boldsymbol{\beta}^\top \mathbf{x}_n + \beta_0)$$

This is so because if $Z \sim Ber(p)$, then $\mathbb{E}[Z] = p$

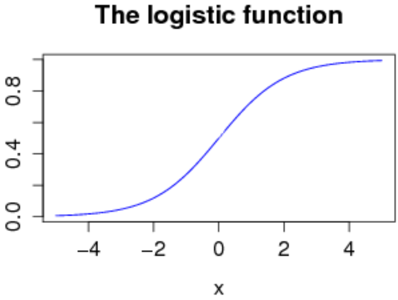
---

obviously $P(\omega_2|\mathbf{x}) = 1 - P(\omega_1|\mathbf{x}) = 1 - y(\mathbf{x})$

Logistic regression

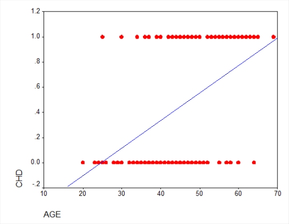
A convenient "interface" function is

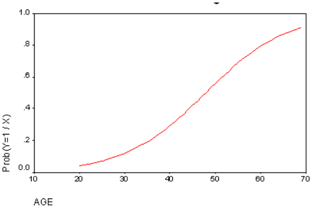
$$g^{-1}(z) = \frac{\exp(z)}{1+\exp(z)} = \frac{1}{1+\exp(-z)}, \qquad \textbf{the logistic function}$$



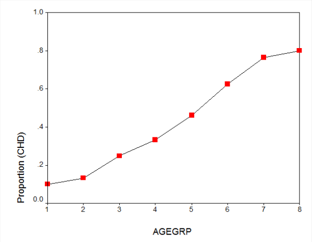
It is a C^∞ function $\mathbb{R} \longrightarrow (0,1)$, and a bijection (one-to-one), with inverse $g(z) = \ln\left(\frac{z}{1-z}\right)$ for $z \in (0,1)$ (the **logit function**)

Logistic regression





Age Group	n	CHD absent	CHD present	Mean (Proportion)
20 – 29	10	9	1	0.10
30 – 34	15	13	2	0.13
35 – 39	12	9	3	0.25
40 – 44	15	10	5	0.33
45 – 49	13	7	6	0.46
50 – 54	8	3	5	0.63
55 - 59	17	4	13	0.76
60 - 69	10	2	8	0.80
Total	100	57	43	0.43



Interpretation of the Logistic regression (I)

The Logistic regression mantra

"The log of the odds is a linear function of the predictors"

Since $P(\omega_1|\mathbf{x}) = g^{-1}(\boldsymbol{\beta}^\top \mathbf{x} + \beta_0)$, we have

$$\ln\left(\frac{P(\omega_1|\mathbf{x})}{P(\omega_2|\mathbf{x})}\right) = \ln\left(\frac{P(\omega_1|\mathbf{x})}{1 - P(\omega_1|\mathbf{x})}\right) = \mathrm{logit}(P(\omega_1|\mathbf{x})) = \boldsymbol{\beta}^\top \mathbf{x} + \beta_0$$

Interpretation of the Logistic regression (II)

$$\text{logODDS}(\mathbf{x}_0) = \ln\left(\frac{P(\omega_1|\mathbf{x}_0)}{P(\omega_2|\mathbf{x}_0)}\right) = \boldsymbol{\beta}^\top \mathbf{x}_0 + \beta_0$$

$$\Rightarrow \text{ODDS}(\mathbf{x}_0) = \frac{P(\omega_1|\mathbf{x}_0)}{P(\omega_2|\mathbf{x}_0)} = \exp(\boldsymbol{\beta}^\top \mathbf{x}_0 + \beta_0)$$

Define $\mathbf{1}_i := (0, \ldots, \overset{i)}{1}, \ldots, 0)^\top$ and so
$\mathbf{x}_0 + \mathbf{1}_i = (x_{01}, \ldots, x_{0i} + 1, \ldots, x_{0N})^\top$

$$\Longrightarrow \frac{\text{ODDS}(\mathbf{x}_0 + \mathbf{1}_i)}{\text{ODDS}(\mathbf{x}_0)} = \exp\left((\boldsymbol{\beta}^\top(\mathbf{x}_0 + \mathbf{1}_i - \mathbf{x}_0)\right) = \exp(\beta_i)$$

Logistic regression as a GLM

For the logistic regression the link function is the logit and the suitable distribution is the Bernoulli.

- As for linear regression, we set the problem as a **Maximum Likelihood** problem with parameters $\boldsymbol{\beta}$ and β_0
- In this case there is no closed-form solution and we use an iterative **Newton-Raphson** method
- This leads to an iterative reestimation procedure for the $\boldsymbol{\beta}$ parameters (IRLS) $\longrightarrow \hat{\boldsymbol{\beta}}$

The Deviance and the AIC

In the context of Generalized Linear Models,

$$-2l(\hat{\beta}) = -2 \ln \mathcal{L}(\hat{\beta})$$

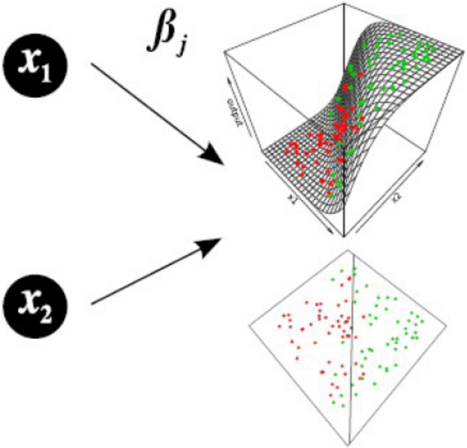
is called the **deviance** (in ML, this is the **error**)

Null deviance: deviance of the null model (just with constant term)
Residual deviance: deviance of the proposed model

AIC

The AIC complements the deviance with complexity penalization $-2l(\hat{\beta}) + 2d$ (a form of **regularization**)

A graphical view of the logistic regression



Exercise: the Bank Marketing dataset

Direct marketing campaign (by means of phone calls) from a Portuguese banking institution

- Often, more than one contact to the same client was required, in order to access if the product would be subscribed
- Number of Instances: 45,211 and 16 predictors, of very different nature and type, including factors, '999' and 'unknown'
- The **target** variable is whether a term deposit was subscribed ('yes') or not ('no')

Play with the dataset to find the best possible predictive model and deliver a **two-page** pdf with what you finally did and what you got

---

More information can be found in https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

Exercise: Bank Marketing dataset

The **input** (predictive) variables are:

1. bank client data:

```
1. age (numeric)
2. job : type of job
     ("admin.","unknown","unemployed","management","housemaid","entrepreneur",
      "student", "blue-collar","self-employed","retired","technician","services")
3. marital : marital status ("married","divorced","single")
4. education ("unknown","secondary","primary","tertiary")
5. default: has credit in default? ("yes","no")
6. balance: average yearly balance, in euros (numeric)
7. housing: has housing loan? ("yes","no")
8. loan: has personal loan? ("yes","no")
```

2. related with the last contact of the current campaign:

```
 9. contact: contact communication type ("unknown","telephone","cellular")
10. day: last contact day of the month (numeric)
11. month: last contact month of year ("jan", "feb", "mar", ..., "nov", "dec")
12. duration: last contact duration, in seconds (numeric)
```

3. other variables:

```
13. campaign: number of contacts performed during this campaign and for this client
     (numeric, includes last contact)
14. pdays: number of days passed after the client was last contacted from a previous campaign
     (numeric, -1 means client was not previously contacted)
15. previous: number of contacts performed before this campaign and for this client (numeric)
16. poutcome: outcome of the previous marketing campaign ("unknown","other","failure","success")
```