Machine Learning: Business Applications

Francisco Rosales Marticorena, PhD.
frosales@esan.edu.pe

04.04.19 – 23.05.19

ESAN Graduate School of Business

Datos Generales del Curso

Asignatura:	Machine Learning: Aplicaciones en los Negocios
Área académica:	Programa de Especialización para Ejecutivos
Año y semestre:	2019 – I
Profesor:	Francisco Rosales Marticorena, PhD.
	Mail: frosales@esan.edu.pe
	Teléfono: (511) 317-7200 / 444340

Sumilla

- Este curso presenta métodos de machine learning con énfasis en problemas de clasificación y regresión de aprendizaje supervisado.
- El curso contiene sesiones de fundamentos matemáticos; sesiones de desarrollo metodológico y de aplicaciones.
- Se usará el software R para la resolución de casos de estudio.

Objetivos de la Asignatura

- Mejorar las capacidades cuantitativas de analistas y gerentes para interpretar los resultados de métodos que aprenden de los datos.
- Utilizar adecuadamente conceptos matemáticos básicos involucrados en los métodos de aprendizaje supervisado de Machine Learning.
- Utilizar el software R y sus librerías especializadas para es desarrollo de implementaciones propias o de terceros.

Programación de Contenidos

1. Estadística básica:
   - Sesión 1: Introducción
   - Sesión 2: Software R
   - Sesión 3: Regresión lineal
2. Métodos Lineales:
   - Sesión 4: Modelos de Clasificación
   - Sesión 5: Métodos de Resampleo
   - Sesión 6: Regularización
   - Sesión 7: Reducción dimensional
   - Sesión 8: Taller
3. Métodos No-Lineales:
   - Sesión 9: Splines
   - Sesión 10: GAMs
   - Sesión 11, 12: Árboles de decisión
   - Sesión 13, 14: Support Vector Machines
   - Sesión 15: Evaluación Final

Metodología

Las exposiciones del profesor se complementarán con actividades que harán los alumnos en el salón de clase, y fuera de él:

- Participar en clase.
- Leer la bibliografía indicada en el programa.
- Hacer las tareas.
- Rendir las evaluaciones programadas.

Evaluación

Nota Final $=$ seis tareas (60%) $+$ un examen final (40%).

- Las tareas se podrán realizar de manera individual o en parejas.
- El examen final es individual.
- El examen final es obligatorio, y se rendirá el día 23.05.19.

Fuentes de Información

- [EH06] Everitt, B. and T. Hothorn. A handbook of statistical analyses using R. Chapman & Hall/CRC, 2006.
- [JO13] James, G. et. al. (2013). An Introduction to Statistical Learning with Applications in R. Springer Series in Statistics.
- [HT09] Hastie, T., R. Tibshirani and J. Friedman (2009). The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer Series in Statistics.
- [RS05] Ramsay, J.O. and B.W. Silverman (2006). Functional Data Analysis. Springer Series in Statistics.
- [RO03] Ruppert, D., M. Wand and R. Carrol (2003). Semiparametric Regression. Cambridge Series in Statistical and Probabilistic Mathematics.
- [W06] Wood, S. (2006). Generalized Additive Models: An Introduction with R. Chapman & Hall/CRC.

Docente

Educación:

- Doctor. Matemáticas y Ciencia Comp. Universidad de Göttingen.
- Magister. Matemáticas Ap. y Estadística. SUNY Stony Brook.
- Magister. Matemáticas. PUCP.
- Licenciado y Bachiller. Economía. UP.

Experiencia:

- 2019: Profesor Investigador. TI. U. ESAN.
- 2018: Gerente. Financial Services Office. EY Perú.
- 2017: Profesor investigador. Finanzas. U. del Pacífico.
- 2011–2016: Investigador asociado. IMS. U. Göttingen.
- 2005–2008: Científico Investigador. CGIAR. CIP.

Asistentes

- Expectativa: analista, gerente, etc.
- Sectores: banca, seguros, reguladores, etc.
- Lenguajes: Python, R, C++, Matlab, etc.

Materiales

Desde un celular:

Desde un navegador:

`https://github.com/LFRM/Lectures`

Introduction

Overview

- Machine Learning: is a toolbox to understand data using statistics
- Objectives
  1. Prediction / to predict something
  2. Inference / to explain something
- Problems
  1. Supervised learning: "input $\Rightarrow$ output" structure.
  2. Unsupervised learning: only "input" structure.
- Methods
  1. Regression
  2. Classification
  3. Clustering

Basics

The general model follows:

$$Y = f(X) + \epsilon, \quad X = \{X_1, \ldots, X_p\}$$

where

- Y is called response / dependent variable
- X are called features / independent variables / predictors
- f is a non-random function
- ϵ is a random error term, independent of X, and with zero mean.

In this course: we find f to predict / explain.

Basics

Usual Steps:

1. We observe predictors X and response Y.
2. We characterize their relationship

$$Y = f(X) + \epsilon. \tag{1}$$

3. We estimate $\hat{f}$ "somehow".
4. We use $\hat{f}$ in X to make prediction $\hat{Y}$

$$\hat{Y} = \hat{f}(X). \tag{2}$$

Focus depends on Goals:

- Prediction: we care mostly about $\hat{Y}$.
- Inference: we care mostly about $\hat{f}$.

Estimation Error

Proposition 1

$$\mathbb{E}[(Y - \hat{Y})^2] = \underbrace{\mathbb{E}[(f(X) - \hat{f}(X))^2]}_{\textit{Reducible}} + \underbrace{\text{Var}[\epsilon]}_{\textit{Irreducible}} . \tag{3}$$

Proof. Trivial. Direct substitution from (1) and (2). ■

Interpretation of proposition 1:

- The magnitude of the estimation error has a reducible and an irreducible component.
- We cannot reduce the estimation error below $\text{Var}[\epsilon]$.

Estimation of f: Parametric vs. Non-Parametric

Parametric Methods:

- Impose rigid structure on f, e.g. f is linear.
- Trade-off: easy to interpret vs. bad accuracy.

Non-Parametric Methods:

- Impose flexible structure on f, e.g. f is a piecewise polynomial.
- Trade-off: difficult to interpret vs. good accuracy.

Estimation of f: Parametric vs. Non-Parametric

Source: [JO13]

Estimation of f: Regression vs. Classification

- Discrete response $\Rightarrow$ classification / cont. response $\Rightarrow$ regression.
- Specific methods for regression or classification.
- Some methods deal with both, e.g. K-nearest neighbors, boosting.

Estimation of f: Assessing Model Accuracy

Definition 1.1 (Mean Squared Error)

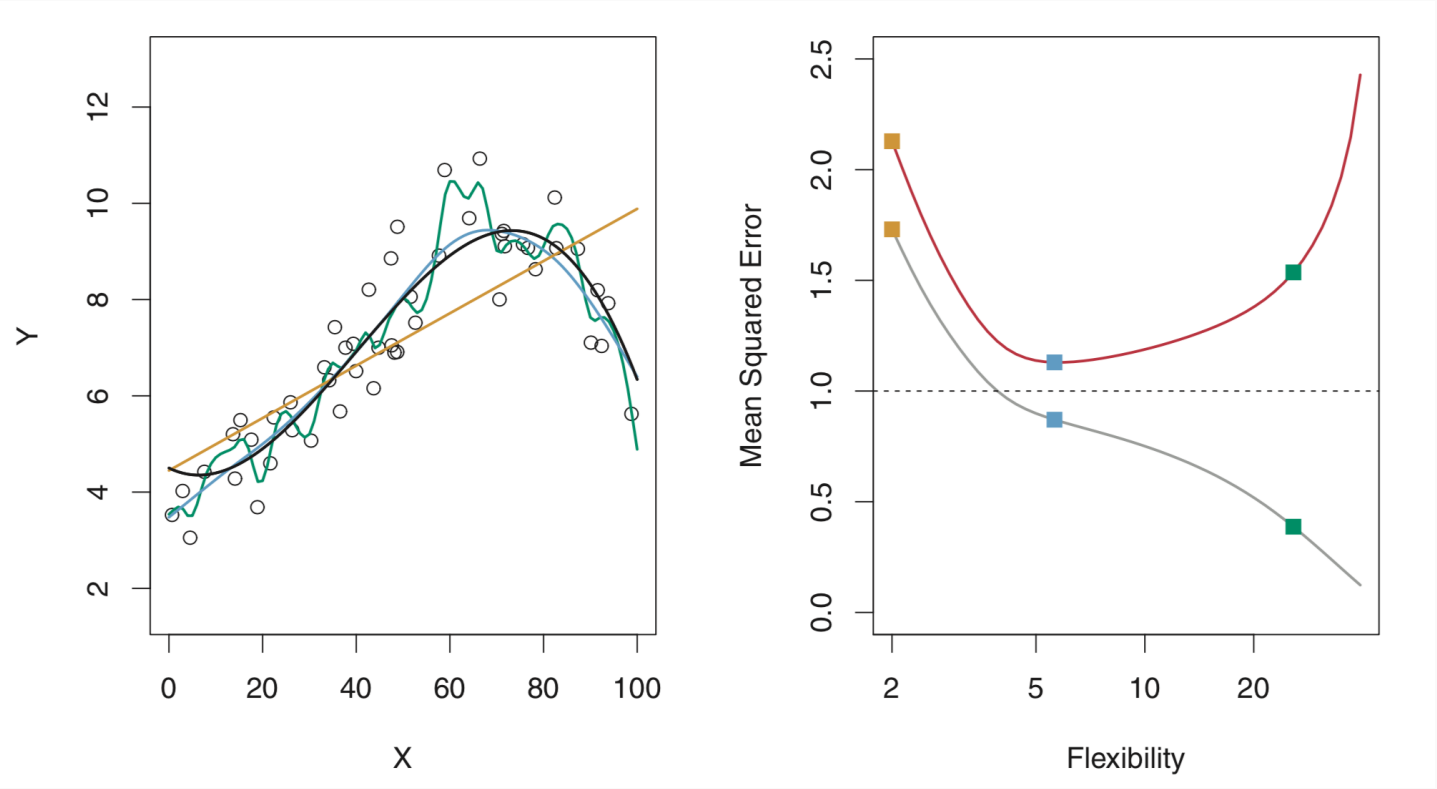
The Mean Squared Error (MSE) is defined as

$$\begin{aligned} MSE &:= Ave\{(y_i - \hat{f}(x_i))^2\} \\ &= \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2. \end{aligned} \tag{4}$$

We call it "train" MSE if we compute it with training data (X, Y) and "test" MSE if we compute it with "test" data (X_0, Y_0).

- Train MSE can be reduced arbitrarily (overfitting).
- Test MSE cannot be reduced arbitrarily.
- Test MSE is used for model selection.

Estimation of f: Assessing Model Accuracy



Left: data (circles), true function (black), linear fit (orange), spline fit 1 (blue), spline fit 2 (green). Right: train MSE (gray), test MSE (red).

Source: [JO13]

Estimation of f: Bias - Variance Trade-off

Proposition 2

Given x_0, the expected test MSE can be decomposed into the sum of three quantities: the variance of $\hat{f}(x_0)$, the squared bias of $\hat{f}(x_0)$ and the variance of the error term.

Proof. From proposition 1 we known that

$$\begin{aligned}\mathbb{E}[(y_0 - \hat{f}(x_0))^2] &= \mathbb{E}[(f(x_0) - \hat{f}(x_0))^2] + \text{Var}[\epsilon] \\ &= \mathbb{E}[(\{f(x_0) - \mathbb{E}[\hat{f}(x_0)]\} \\ &\quad -\{\hat{f}(x_0) - \mathbb{E}[\hat{f}(x_0)]\})^2] + \text{Var}[\epsilon],\end{aligned}$$

Thus,

$$\begin{aligned}\mathbb{E}[(y_0 - \hat{f}(x_0))^2] &= \underbrace{\mathbb{E}[(f(x_0) - \mathbb{E}[\hat{f}(x_0)])^2]}_{\text{Squared Bias of } \hat{f}} + \underbrace{\mathbb{E}[(\hat{f}(x_0) - \mathbb{E}[\hat{f}(x_0)])^2]}_{\text{Variance of } \hat{f}} \\ &\quad + \underbrace{\text{Var}[\epsilon]}_{\text{Variance of error}}\end{aligned}$$

Estimation of f: Bias - Variance Trade-off

Interpretation of proposition 2:

- The reducible error is a mixture of bias and variance.
- To minimize the expected test MSE we need a method that minimizes both Bias and Variance.

Methods:

- More flexible methods: $\downarrow$ bias and $\uparrow$ variance.
- Less flexible methods: $\uparrow$ bias and $\downarrow$ variance.
- The adequacy of the method depends on the data.

Estimation of f: Classification Setting

Definition 1.2 (Training Error Rate)

The training error rate (TER) is the proportion of mistakes that the classifier makes in the training set

$$\begin{aligned} TER: &= Ave\{\mathcal{I}_{\hat{y}_i \neq y_i}\} \\ &= \frac{1}{n}\sum_{i=1}^{n} \mathcal{I}_{\hat{y}_i \neq y_i}, \end{aligned} \tag{5}$$

Definition 1.3 (Bayes Classifier)

The Bayes classifier minimizes (5), by selecting the value of j such that

$$max_j \mathbb{P}[Y = j | X = x_0], \tag{6}$$

for each x_0 in the test data set.

Estimation of f: Classification Setting

Note that the Bayes classifier:

- Is theoretical, i.e. the conditional probability in (6) is unknown.
- Provides a lower bound on TER,

$$\text{TER} \geq 1 - \mathbb{E}\left[\max_j \mathbb{P}[Y = j | X = x_0]\right]$$

Estimation of f: Classification Setting

Definition 1.4 (K-Nearest Neighbors)

K-Nearest Neighbors (KNN) is a classification method that requires a positive integer K. For a test observation x_0, it identifies K points near x_0, called $\mathcal{N}_0$, and estimates the conditional probability for class j as:

$$\hat{\mathbb{P}}[Y = j | X = x_0] = \frac{1}{K} \sum_{i \in \mathcal{N}_0} \mathcal{I}_{y_i = j},$$

i.e. as the fraction of the points in $\mathcal{N}_0$ whose response values equal j.

Note that:

- K has a strong effect on the classification obtained by KNN.
- Small K: $\downarrow$ bias and $\uparrow$ variance.
- Large K: $\uparrow$ bias and $\downarrow$ variance.

R Software

The R Project for Statistical Computing

- This is an extremely short introduction to R.
- Click here to download the software.
- For more information look at my lecture notes in Applied Statistics.

For now, lets run R online:

Basic Commands

- The call for the `funcname` function is `funcname(arg1,arg2)`, where `arg1` and `arg2` are args. of the function.
  1. What does `c()` evaluated in the arguments 1 and 2 does?
  2. Does the order of the arguments matter?
- We can store values using "<-" or "=".

```
> x <- c(1,2)
> y = c(4,-1)
> x + y
[1]  5 1
```

Let `z1 = 1000` and `z2 = rep(1000, 3)`. Compute

1. `x + z1`
2. `x + z2`

Basic Commands

- To ask for help on function `funcname` use `?funcname`
- To list all the objects in the workspace use `ls()`.
- To delete object `obj` from the workspace use `rm(obj)`.
- To delete every object use `rm(list = ls())`

Basic Commands

- To create a matrix use `matrix`

```
> x <- matrix(data = 1:4, nrow = 2, ncol = 2)
> x
     [,1] [,2]
[1,]    1    2
[2,]    3    4
```

- Compute
  1. `z <- x ^ 2`
  2. `r <- matrix(1, 2, 4)`
  3. `q <- z %*% r`

Basic Commands

- To create a realization of a normal random variable use `rnorm`. To specigy the mean, use `mean` and, to specify the standard deviation, use `sd`

```
> x <- rnorm(10000)
> y <- rnorm(10000, mean = 50, sd = 0.1)
```

- To compute the correlation between two r.v.s use `cor`.

```
> z <- x + y
> cor(x, z)
[1]  0.9951
```

- To reproduce the exact same random number use `set.seed()` with an arbitrar integer argument, e.g. `set.seed(123)`

```
> set.seed(123)
> rnorm(5)
[1]  -0.56047565 -0.23017749  1.55870831  ...
```

Graphics

- There are various functions for plotting. See `?plot`

```
> x = rnorm(100) + 1:100
> y = rnorm(100) + seq(-1, -100, length = 100)
> plot(x, y)
> plot(x, y, xlab = "this is my x-axis",
  ylab = "this is my y lab", main = "Plot x vs y")
```

- To save the output use `pdf()`, or `jpeg()`

```
> pdf("myfgure.pdf")
> plot(x, y, color = 2, lwd = 3)
> dev.off()
null device
          1
```

Graphics

Let $f : \mathbb{R}^2 \to \mathbb{R}$,

- To plot the contour of f use `contour` or `image`.

```
> x = 1:10
> y = x
> f = outer( x, y, function (x, y) cos(y) / (1 + x ^ 2) )
> contour(x, y, f)
> contour(x, y, f, nlevels = 45)
> fa = ( f - t(f) ) / 2
> contour( x, y, fa, nlevels = 15)
> image(x, y, fa)
```

- To plot f in three dimensions use `persp`.

```
> persp(x, y, fa)
> persp(x, y, fa, theta = 30, phi = 20)
> persp(x, y, fa, theta = 30, phi = 40)
```

Indexing Matrix Data

Extracting part of a data set can be done in different ways:

```
> A = t(matrix(1:16, 4, 4))
> A
     [,1] [,2] [,3] [,4]
[1,]    1    2    3    4
[2,]    5    6    7    8
[3,]    9   10   11   12
[4,]   13   14   15   16
```

Compute

1. `A[2,3]`
2. `A[1,]`
3. `A[1:3, c(2,4)]`
4. `A[,-4]`
5. `dim(A)`

Loading Data

- To import data use `read.table()` or `read.csv()`; and to visualize the imported data use `fix()`.

```
> Auto = read.csv("Auto.csv", header = T,  sep=" ")
> fix(Auto)
> dim(Auto)
[1] 392 9
```

- To list the variable names in the data set use `names()`.

```
> names(Auto)
[1] "mpg"     "cylinders"     "displacement" "horsepower"
[5] "weight"  "acceleration"  "year"         "origin"
[9] "name"
```

- To export data use `write.table()`

Additional Graphical and Numerical Summaries

- To access a variable `cylinders` in data frame `Auto`, we use

  ```
  > Auto$cylinders
  ```

- To avoid using the dollar symbol, we can simply attach the data, so that all the variables in the data frame are added to the workspace.

  ```
  > attach(Auto)
  > plot(cylinders, mpg)
  ```

- Plot the following:
  1. mpg vs. cylinders using red circles
  2. mpg vs. cylinders using a red circles, with axis labels "mpg" and "cylinders" resp.

Additional Graphical and Numerical Summaries

- To plot a histogram use `hist()`

```
> hist(mpg)
> hist(mpg, col = 2)
```

- To create a scatterplot matrix use `pairs()`

```
> pairs(Auto)
> pairs(~ mpg + displacement + horsepower
  + weight + acceleration, Auto)
```

- To print a summary of a given variable or data frame, use `summary()`

```
> summary(mpg)
> summary(Auto)
```

Homework

Exercises 2, 3, 4, 6, 7, 9, 10 from [JO13].

Linear Regression

Motivation

What?

Supervised learning method for continuous response.

Why?

- Well document starting point.
- Recall: not flexible, tipically $\uparrow$ bias and $\downarrow$ variance.
- Fancy approaches are extensions/generalizations of linear regression.

Simple Linear Regression

Model:

$$Y = f(X) + \epsilon, \quad f(X) = \beta_0 + \beta_1 X, \tag{7}$$

where ϵ is a centered random noise, uncorrelated to X.

Estimation

- Idea: estimate $f(X)$ via

$$\hat{f}(X) = \hat{\beta}_0 + \hat{\beta}_1 X, \tag{8}$$

  e.g. identifying $\hat{\beta}_0$, $\hat{\beta}_1$ that fulfills the least squares criteria.

- Least Squares Criteria:

$$(\hat{\beta}_0, \hat{\beta}_1) \quad = \quad \text{argmin}_{\hat{\beta}_0, \hat{\beta}_1} \text{RSS} \tag{9}$$

  where the minimized quantity:

$$\text{RSS} := \sum_{i=1}^{n} e_i^2, \quad e_i := y_i - \hat{y}_i, \tag{10}$$

  is the "Residual Sum of Squares" (RSS) of the model.

Estimation

- FOC:

$$\frac{\partial RSS}{\partial \hat{\beta}_0} = \hat{\beta}_0 + \hat{\beta}_1 \bar{x} - \bar{y} = 0, \tag{11}$$

$$\frac{\partial RSS}{\partial \hat{\beta}_1} = \hat{\beta}_0 \sum_{i=1}^{n} x_i + \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i y_i = 0. \tag{12}$$

  with solution

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}, \tag{13}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}. \tag{14}$$

- RSS is convex, thus the SOC is fullfilled.

Unbiasedness

Recall:

- Let $Z \sim (\mu, \sigma)$, where mean μ and s.d. σ are unknown.
- Sample mean estimator: Collect n random samples of Z, and estimate μ by $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} z_i$. Repeat the previous m times with different samples of size n: $\{\hat{\mu}^{(1)}, \hat{\mu}^{(2)}, \ldots, \hat{\mu}^{(m)}\}$.
- Unbiasedness: We say $\hat{\mu}$ is an unbiased estimator of μ if

$$\frac{1}{m}\sum_{k=1}^{m} \hat{\mu}^{(k)} \to \mu \quad \text{as} \quad k \to \infty.$$

- Variance: The precision of the sample mean estimator is

$$\text{Var}[\hat{\mu}] = \text{SE}(\hat{\mu})^2 = \frac{\sigma^2}{n}.$$

Linear regression:

- β_0 and β_1 are r.v.s, and given some data we estimate them by $\hat{\beta}_0$ and $\hat{\beta}_1$, but the estimators vary according to the random sample.
- Unbiasedness: if we consider m random samples, we obtain $\hat{\beta}_0^{(k)}$ and $\hat{\beta}_1^{(k)}$, for $k = 1, \ldots, m$. It can be shown that

$$\frac{1}{m}\sum_{k=1}^{m}\hat{\beta}_0^{(k)} \to \beta_0 \quad \text{and} \quad \frac{1}{m}\sum_{k=1}^{m}\hat{\beta}_1^{(k)} \to \beta_1 \quad \text{as} \quad k \to \infty,$$

  thus $\hat{f}(X)$ is an unbiased estimator of $f(X)$.
- Variance: The precision of the estimators follow:

$$\text{Var}[\hat{\beta}_0] = \sigma^2\left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right], \tag{15}$$

$$\text{Var}[\hat{\beta}_1] = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}. \tag{16}$$