Tecnológico de Monterrey

Clasificación

Dr. Omar Mendoza Montoya

03/09/2020

Clasificación

- En aprendizaje automatizado, clasificación se refiere a la identificación de la categoría a la que una observación pertenece.

¿Es un perro o un gato?

Ejemplos de problemas de clasificación

- Una persona llega a la sala de emergencia presentando un conjunto de síntomas, los cuales se pueden atribuir a tres condiciones médicas. ¿Cuál condición medica sería la que correspondería a los síntomas presentados?

- En un servicio de banca en línea se debe determinar si una transacción realizada fue fraudulenta, tomando como base la IP del usuario, el historial de transacciones, y otras variables.

- De acuerdo con la secuencia de ADN extraída de un grupo de sujetos, de los algunos presentan una enfermedad, un biólogo quisiera determinar qué mutaciones en el ADN son perjudiciales al causar la enfermedad y cuales no.

Clasificación

- Formalmente, un modelo de clasificación es una función que mapea un vector de entrada a una categoría o etiqueta.

$$f(X; W) = \hat{L}$$

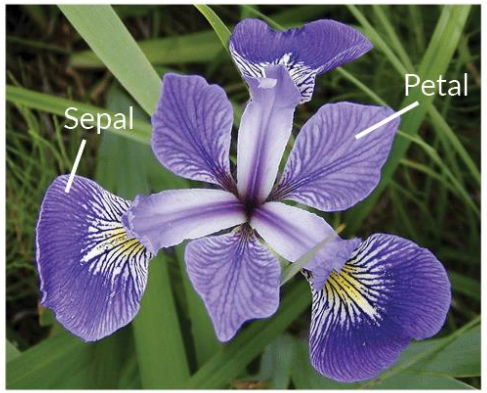
$X =$ Vector de p variables (ordinales, booleanas o categóricas)

$W =$ Parámetros del modelo

$\hat{L} =$ Etiqueta o categoría de la observación X

- Cada variable x_i en el vector X es llamada predictor, característica, dimensión, etc.
- $\hat{L}$ es la etiqueta predicha por el modelo y L es la verdadera etiqueta.
- La pareja (X, L) es una observación o muestra.

Ejemplo: el conjunto de datos Iris



Iris Versicolor

Iris Setosa

Iris Virginica

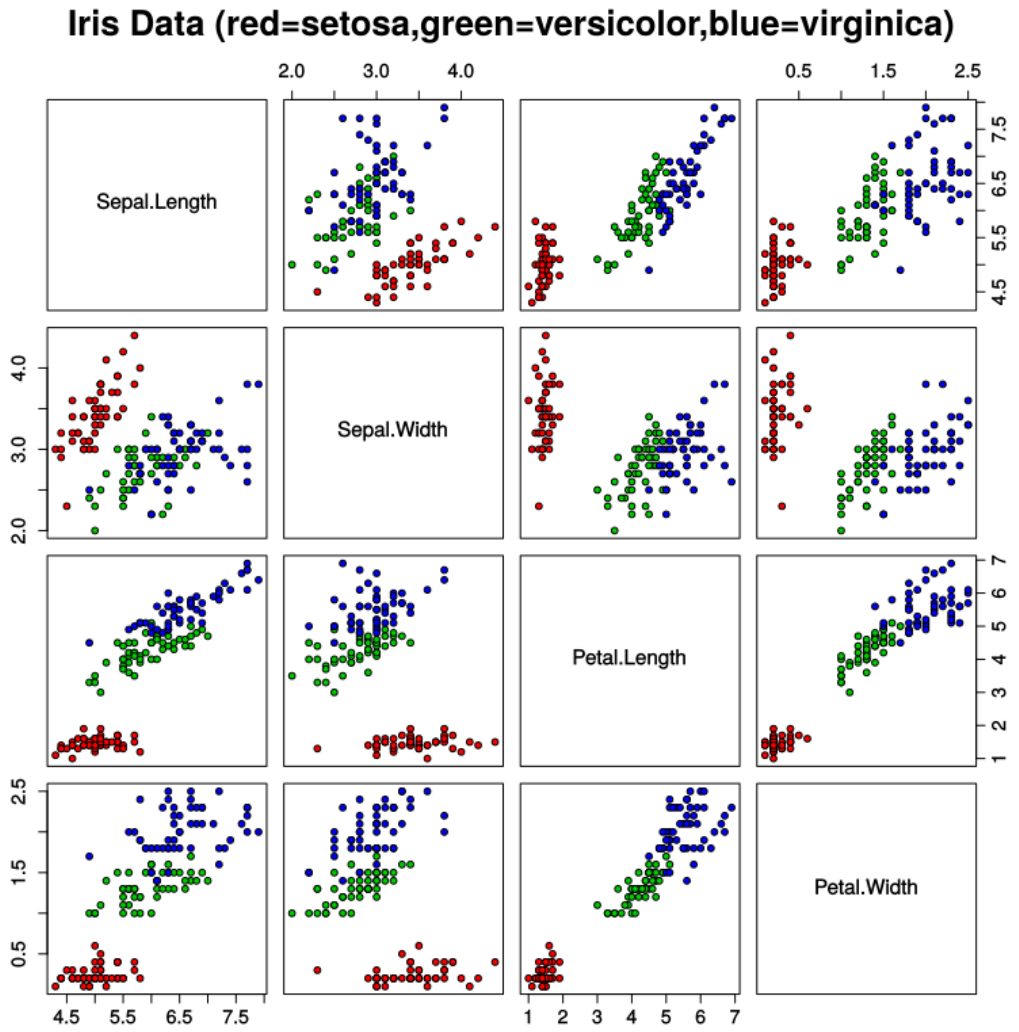
Etiquetas o clases

Observaciones

	Longitud del sépalo	Ancho del sépalo	Longitud del pétalo	Ancho del pétalo	Clase
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
			...		
50	6.4	3.5	4.5	1.2	Versicolor
			...		
150	5.9	3.0	5.0	1.8	Virginica

Predictores, características, variables, espacio de características

Ejemplo: el conjunto de datos Iris



Fuente: Wikipedia

Ejemplo: el conjunto de datos Iris

$$g(X;W) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4$$

$$f(X;W) = \begin{cases} Setosa & \text{si } g(X;W) \text{ se acerca al valor promedio del grupo Setosa} \\ Versicolor & \text{si } g(X;W) \text{ se acerca al valor promedio del grupo Versicolor} \\ Virginica & \text{si } g(X;W) \text{ se acerca al valor promedio del grupo Virginica} \end{cases}$$

$X = [x_1, x_2, x_3, x_4]$ Vector de 4 variables
x_1 = Longitud del sépalo
x_2 = Ancho del sépalo
x_3 = Longitud del pétalo
x_4 = Ancho del pétalo

$W = [w_0, w_1, w_2, w_3, w_4]$ Parámetros del modelo (valores constantes)
$L \in \{Setosa, Versicolor, Virginica\}$ Categorías

Ejemplo: el conjunto de datos Iris

- Independientemente del modelo de clasificación, requerimos de un conjunto de datos de entrenamiento para calcular los parámetros W del modelo.

- Dado un conjunto de entrenamiento con N observaciones $\{(X_1, L_1), (X_2, L_2), \ldots, (X_N, L_N)\}$, el algoritmo de entrenamiento calcula el vector de parámetros W de tal forma que $f(x; W)$ se comporta lo mejor posible para los datos de entrenamiento.

- Por ejemplo, hay modelos para los cuales se resuelve el siguiente problema de optimización:

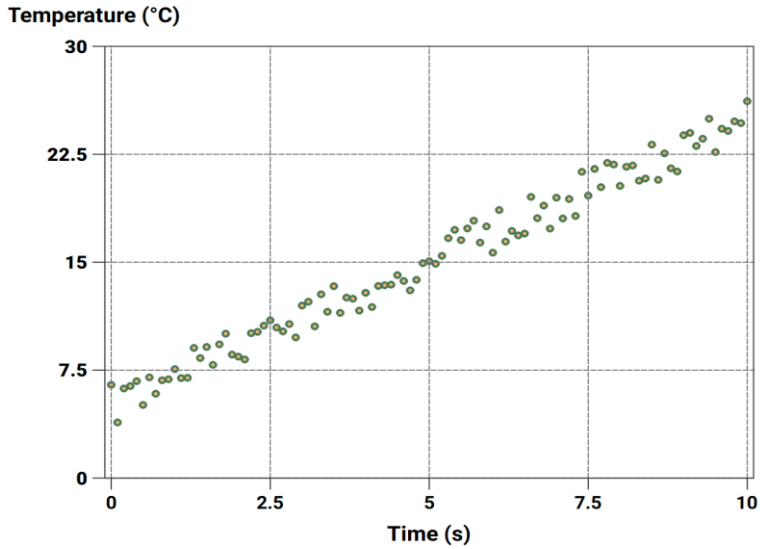
$$W^* = \arg\min_W \frac{1}{N} \sum_{i=1}^{N} I(f(X_i; W) \neq L_i)$$

donde $I(T)$ es la función indicadora, y regresa 1 si T es verdadero, o 0 si T es false.

- Al proceso de encontrar W se le conoce como "aprendizaje supervisado".

Clasificación Vs Regresión

- En regresión, se busca encontrar la relación entre un conjunto de variables de entrada y de salida. Las variables de salida típicamente son continuas. En clasificación, la variable de salida es una etiqueta o categoría.



$$T(t) = at + b$$

- Formalmente, un modelo de regresión es una función que mapea un vector de entrada con un vector de salida:

$$f(X; W) = \hat{Y}$$

X = Vector de entrada de p variables (ordinales o categóricas)

W = Parámetros del modelo

$\hat{Y}$ = Vector de salida de q variables (ordinales)

- El problema consiste en encontrar W de tal forma que $f(X; W)$ explique los datos de la mejor manera posible.

Aspectos importantes en clasificación

- En clasificación, es necesario resolver los siguientes problemas:
  - Es necesario seleccionar un modelo de entre muchas posibilidades (selección de modelo).
  - Después de seleccionar un modelo (típicamente, una función parametrizada), los parámetros del modelo W deben ser estimados (entrenamiento del modelo).
  - Una vez que se tiene un modelo candidato, es necesario determinar su habilidad para predecir correctamente las etiquetas de observaciones que no hayan sido utilizadas en el entrenamiento (evaluación del modelo).

Modelos de clasificación

- Modelos lineales. Análisis discriminante lineal (LDA), máquinas de soporte vectorial (SVM), clasificador de Fisher, clasificador bayesiano ingenuo lineal, clasificador logístico.

- Modelos cuadráticos. Análisis discriminante cuadrático (QDA), clasificador bayesiano ingenuo cuadrático.

- Modelos no lineales. k- vecinos más cercanos (k-NN), análisis discriminante generalizado, maquinas de soporte vectorial de base radial (RBSVM), ADA-boost, redes neuronales, árboles de decisión.

Modelos lineales

- Están basados en el valor que se obtiene por una combinación lineal de las variables predictoras.

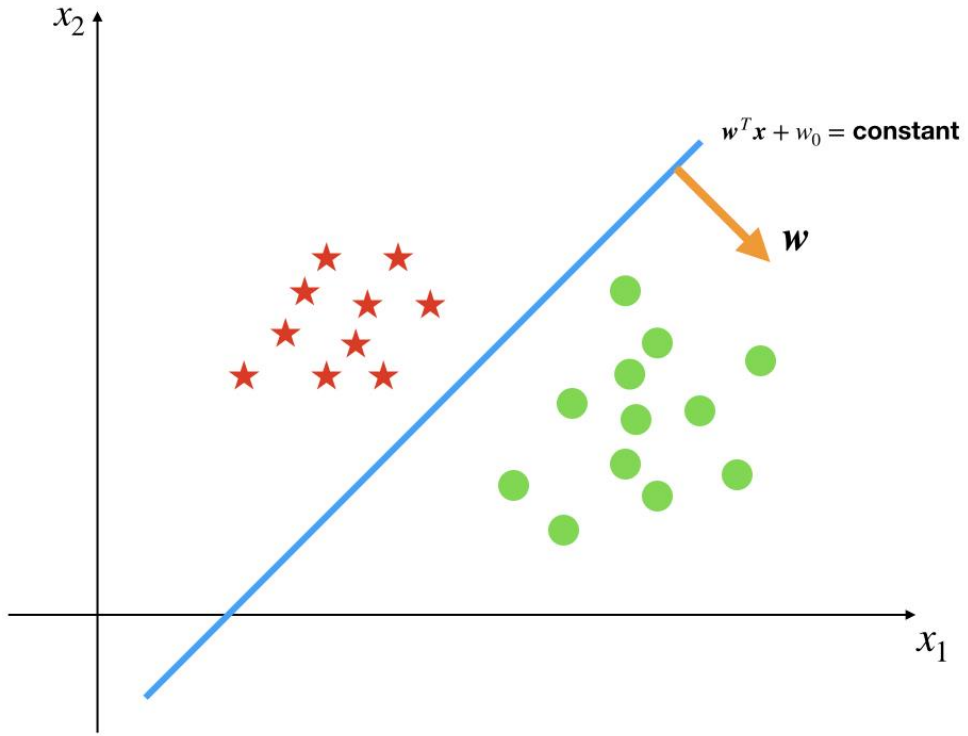
- Para el problema de dos clases, la forma general de un modelo de clasificación lineal está dada por la siguiente expresión:

$$f(X;W) = \begin{cases} 1 & \text{if } g\left(w_0 + \sum_{j=1}^{p} w_j x_j\right) > T \\ 0 & \text{en otro caso} \end{cases}$$

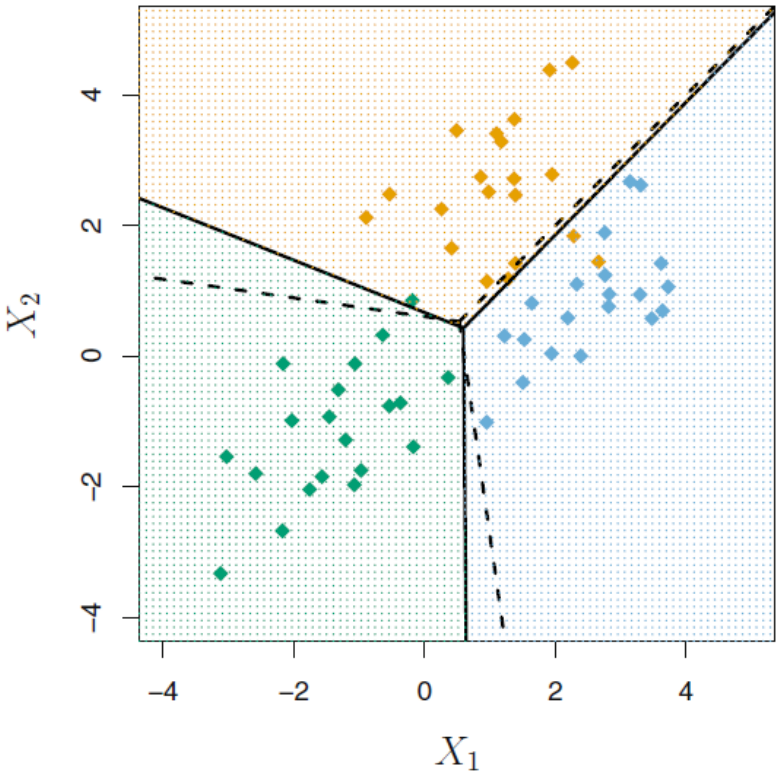
donde T es un valor de umbral.

- La función $g(x)$ puede ser usada con una medida de confianza. Entre mayor sea su valor, hay más evidencia de que la clase sea 1.

Modelos lineales



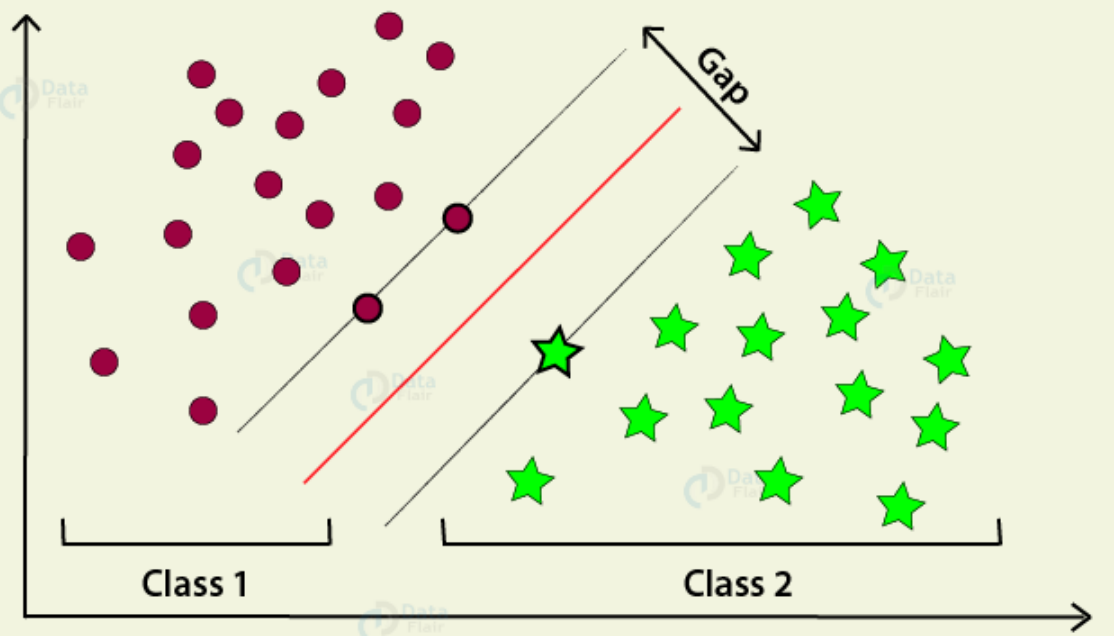
Fuente: https://brainbomb.org/Artificial-Intelligence/Machine-Learning/ML-Linear-Classification-An-Introduction-to-Discriminant-Functions/



Fuente: An introduction to statistical learning: with applications in R.

Máquinas de soporte vectorial (SVM)

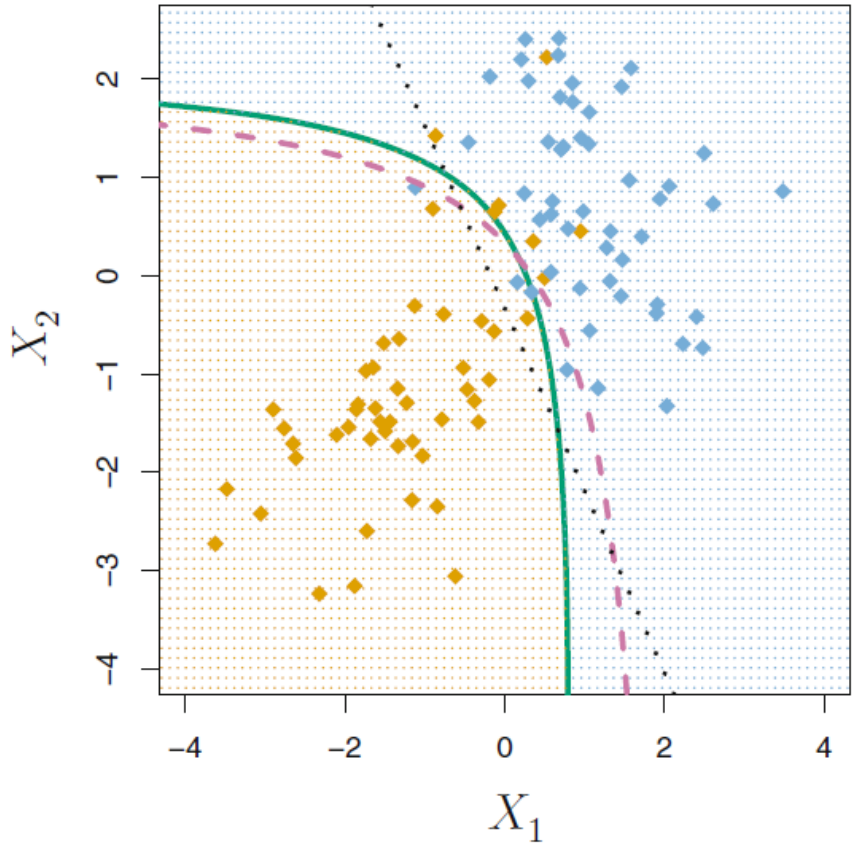
- Este clasificador de dos clases encuentra el hiperplano separador que maximiza la distancia entre las observaciones más cercanas a dicho plano de ambas clase.
- A la distancia entre las observaciones de ambas clases cercanas al hiperplano separador se conoce como margen.



Source: https://data-flair.training/blogs/svm-support-vector-machine-tutorial/

Modelos cuadráticos

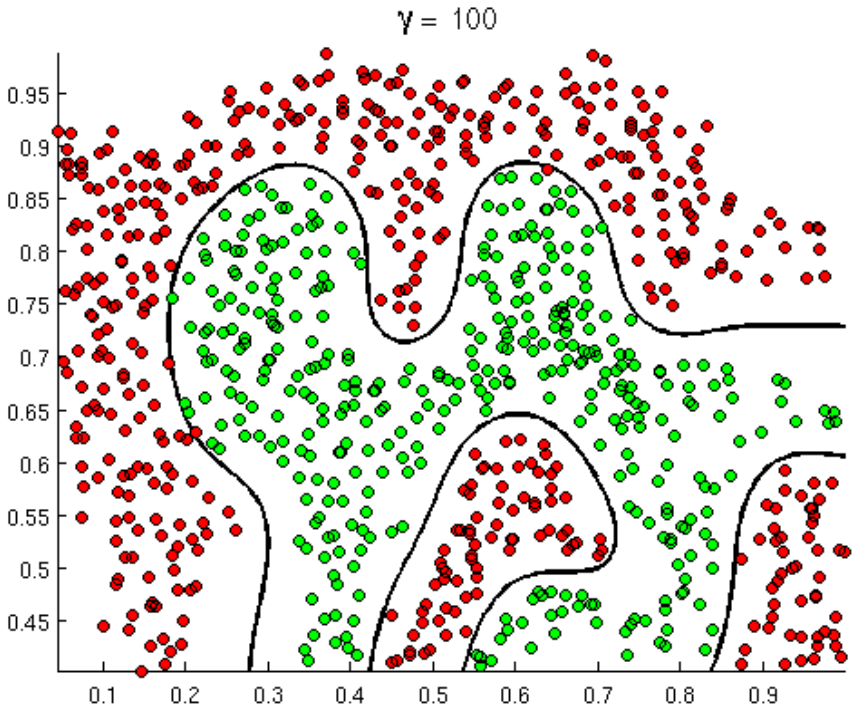
- En estos clasificadores, el límite entre clases está dado por una función cuadrática.



Source: An introduction to statistical learning: with applications in R.

Clasificadores no lineales

- Los límites de decisión no son funciones lineales ni cuadráticas.



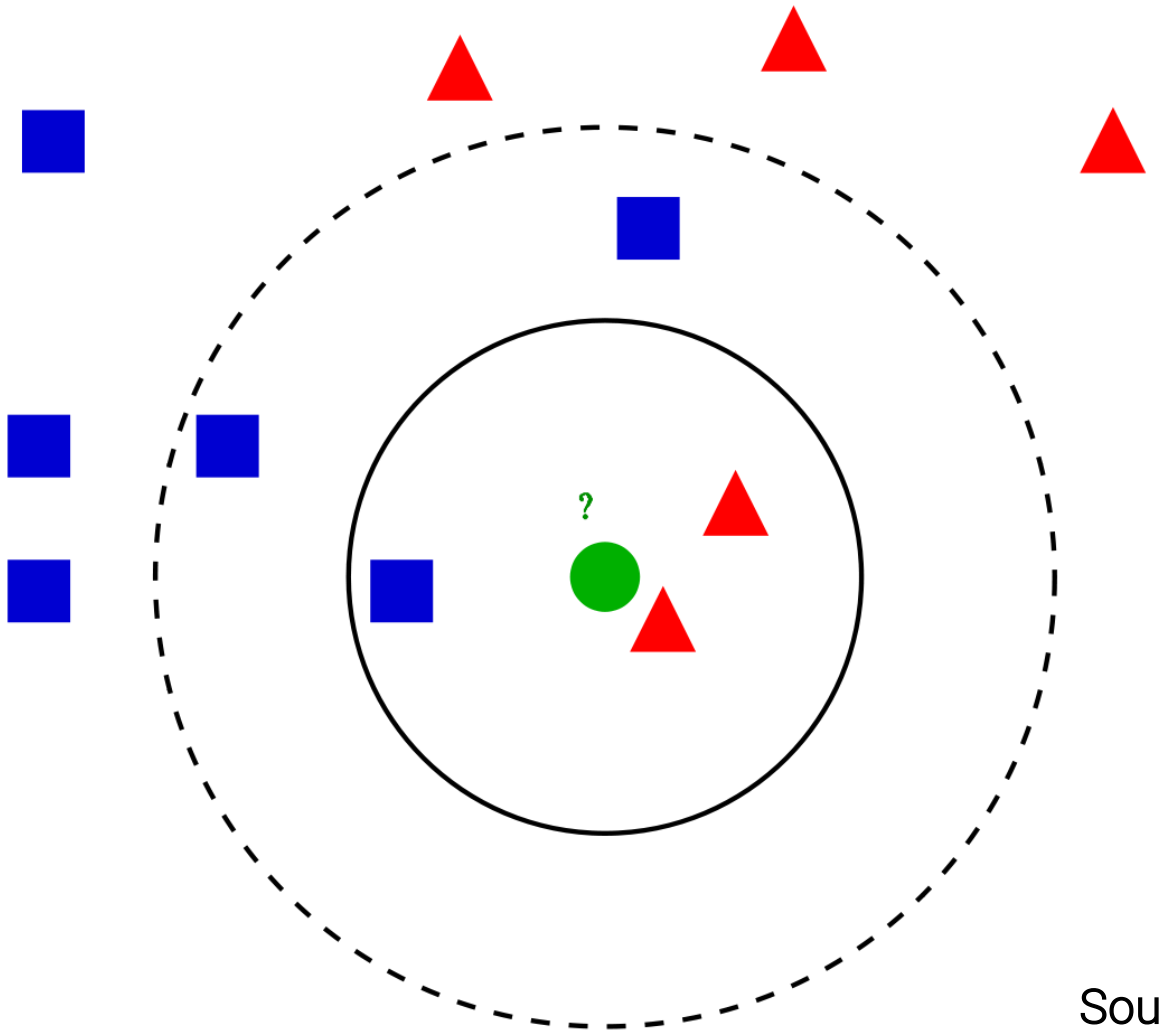
Source:
http://openclassroom.stanford.edu/MainFolder/DocumentPage.php?course=MachineLearning&doc=exercises/ex8/ex8.html

K-vecinos más cercanos (k-NN)

- Método multiclase, en el que se determinan las k observaciones del conjunto de entrenamiento más cercanas a la observación a clasificar.

- Se realiza una votación entre las k elementos más cercanos para determinar la clase de la observación que se está evaluando. Si $k = 1$, la clase simplemente corresponde a la observación más cercana.

K-vecinos más cercanos (k-NN)



Source: Wikipedia.

Clasificación multiclase (más de dos clases)

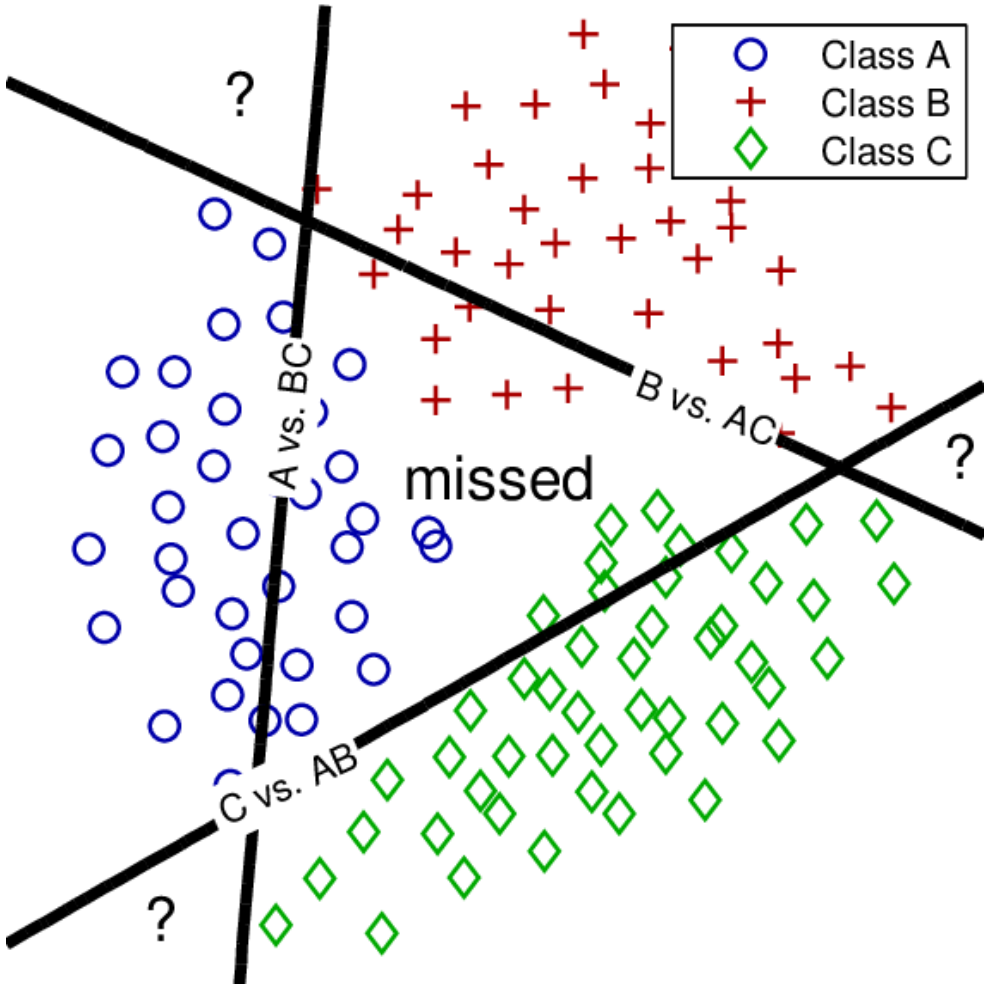
- Hay clasificadores que de manera natural pueden resolver problemas multiclase (k-NN, LDA, redes neuronales, árboles de decisión).

- Sin embargo, muchos clasificadores sólo funcionan con dos clases. Éstos métodos pueden ser utilizados en escenarios multiclase a través de dos estrategias posibles.

- En la estrategia uno contra el resto, se entrenan C clasificadores (donde C es la cantidad de clases), de tal forma que cada clasificador compara las observaciones de una clase contra el resto de observaciones.

- Para determinar la clase de una nueva observación, se evalúan todos los clasificadores. Si sólo uno detecta de manera positiva la clase de la observación, y el resto indica que pertenecen a la clase “resto”, entonces la clase es la del clasificador que dio positivo.

Clasificación multiclase (más de dos clases)



Source:
https://www.researchgate.net/figure/One-versus-Rest-Ensemble-trained-on-the-data-set-from-Fig-24-Each-model-for-class-c_fig9_40220479

Clasificación multiclase (más de dos clases)

- En la reducción uno contra uno, se entrenan $C(C-1)/2$ clasificadores binarios para cada pareja posible de clases ($C_1\ Vs\ C_2$, $C_1\ Vs\ C_3$, $C_2\ Vs\ C_3$, etc.).

- Para predecir la etiqueta de una nueva observación, se evalúan todos los clasificadores, y la clase que reciba más votos, es la que se le asigna a la observación.

Evaluación de modelos de clasificación

Evaluación de modelos

- Dado un modelo de clasificación, se desea estimar su capacidad para predecir correctamente la clase de nuevas observaciones (capacidad de generalización del modelo).

- Para ello, necesitamos un nuevo conjunto de datos (conjunto de prueba), con el cual es posible estimar la exactitud del modelo, así como error de predicción.

- No podemos usar el conjunto de entrenamiento, porque el resultado estaría sesgado. Sería como preguntar en un examen lo mismo que se ha visto en clase sin cambio alguno.

Matriz de confusión

- Una forma de resumir los resultados de la evaluación de un modelo es a través de una matriz de confusión.

	Perro (Real)	No perro (Real)	Total
Perro (Predicha)	5 (TP)	1 (FP)	6
No perro (Predicha)	2 (FN)	2 (TN)	4
Total	7	3	10

- TP – Verdaderos positivos
- FP – Falsos positivos
- FN – Falsos negativos
- TN – Verdaderos negativos

Matriz de confusión

$$\text{Accuracy} = \frac{TP + TN}{\text{Total de observaciones}}$$

$$\text{Recall}_{\text{perro}} = \frac{TP}{TP + FN}$$

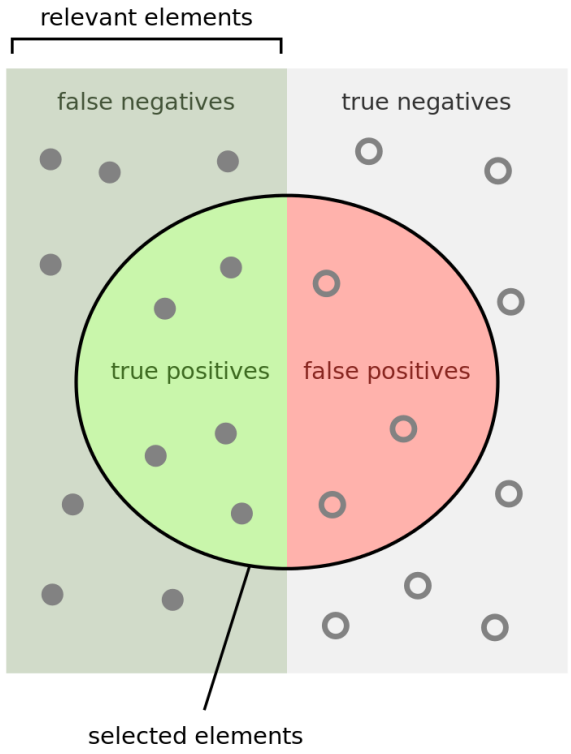
$$\text{Precision}_{\text{perro}} = \frac{TP}{TP + FP}$$

$$\text{Recall}_{\text{no perro}} = \frac{TN}{TN + FP}$$

$$\text{Precision}_{\text{no perro}} = \frac{TN}{TN + FN}$$

$$\text{F} - \text{measure}_i = \frac{2 * \text{Recall}_i * \text{Precision}_i}{\text{Recall}_i + \text{Precision}_i} \quad (Para\ i = \text{perro, no perro})$$

Matriz de confusión



How many selected items are relevant?

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

How many relevant items are selected?

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

Source: Wikipedia

Matriz de confusión

$$\text{Weighted} - \text{Accuracy} = w_P * \frac{TP}{TP + FN} + w_N * \frac{TN}{TN + FP}$$

$$w_P + w_N = 1$$

$$0 \leq w_P \leq 1$$

$$0 \leq w_N \leq 1$$

Matriz de confusión

	Gato (Real)	Pescado (Real)	Gallina (Real)	Total
Gato (Predicha)	4	6	3	13
Pescado (Predicha)	1	2	0	3
Gallina (Predicha)	1	2	6	9
Total	6	10	9	25

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i}$$

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i}$$

TP_i Verdaderos positivos para la clase i

FN_i Falsos negativos para la clase i

FP_i Falsos positivos para la clase i

Matriz de confusión

$$\text{Accuracy} = \frac{TP_1 + TP_2 + TP_3}{\text{Total de observaciones}}$$

$$\text{Weighted} - \text{Accuracy} = w_1 * \frac{T_1}{T_1 + F_1} + w_2 * \frac{T_2}{T_2 + F_2} + w_3 * \frac{T_3}{T_3 + F_3}$$

$$w_1 + w_2 + w_3 = 1$$

$$0 \leq w_1 \leq 1$$

Selección de hiperparámetros

Hiperparámetros

- Algunos modelos tienen parámetros que no pueden ser estimados en el proceso de entrenamiento.
- Por ejemplo, la k del clasificador k-NN.
- A estos parámetros se les conoce como hiperparámetros.
- Para calcular la configuración óptima de hiperparámetros, es necesario evaluar varias configuraciones, y tomar aquella que de los mejores resultados.
- Como esto también puede sesgar la evaluación del clasificador, es necesario otro conjunto de datos llamado conjunto de validación, con el cual se evalúan las diferentes configuraciones del clasificador.

Conjuntos de datos en aprendizaje supervisado

Conjunto de entrenamiento

Para entrenar el modelo y encontrar sus parámetros.

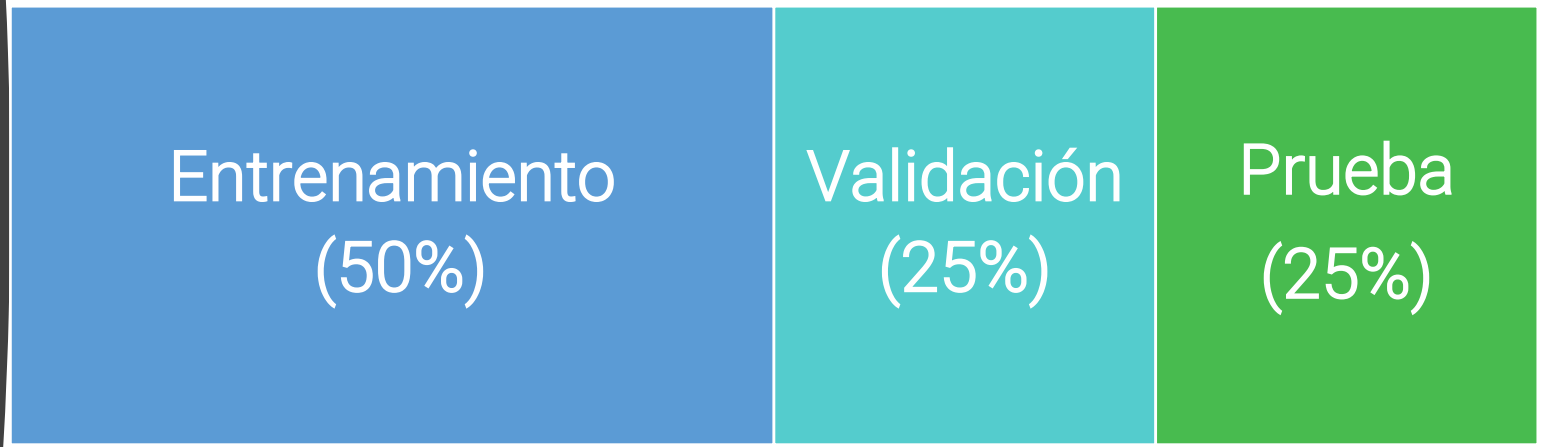
Conjunto de validación

Para encontrar los hiperparámetros del modelo.

Conjunto de prueba

Para estimar el error de clasificación y la previsión del modelo.

Conjuntos de datos en aprendizaje supervisado



¿Qué pasa si mi conjunto de datos es pequeño y no lo podemos dividir en tres?

Validación cruzada

- En la práctica, es difícil tener suficientes observaciones para entrenar adecuadamente el modelo y evaluarlo.

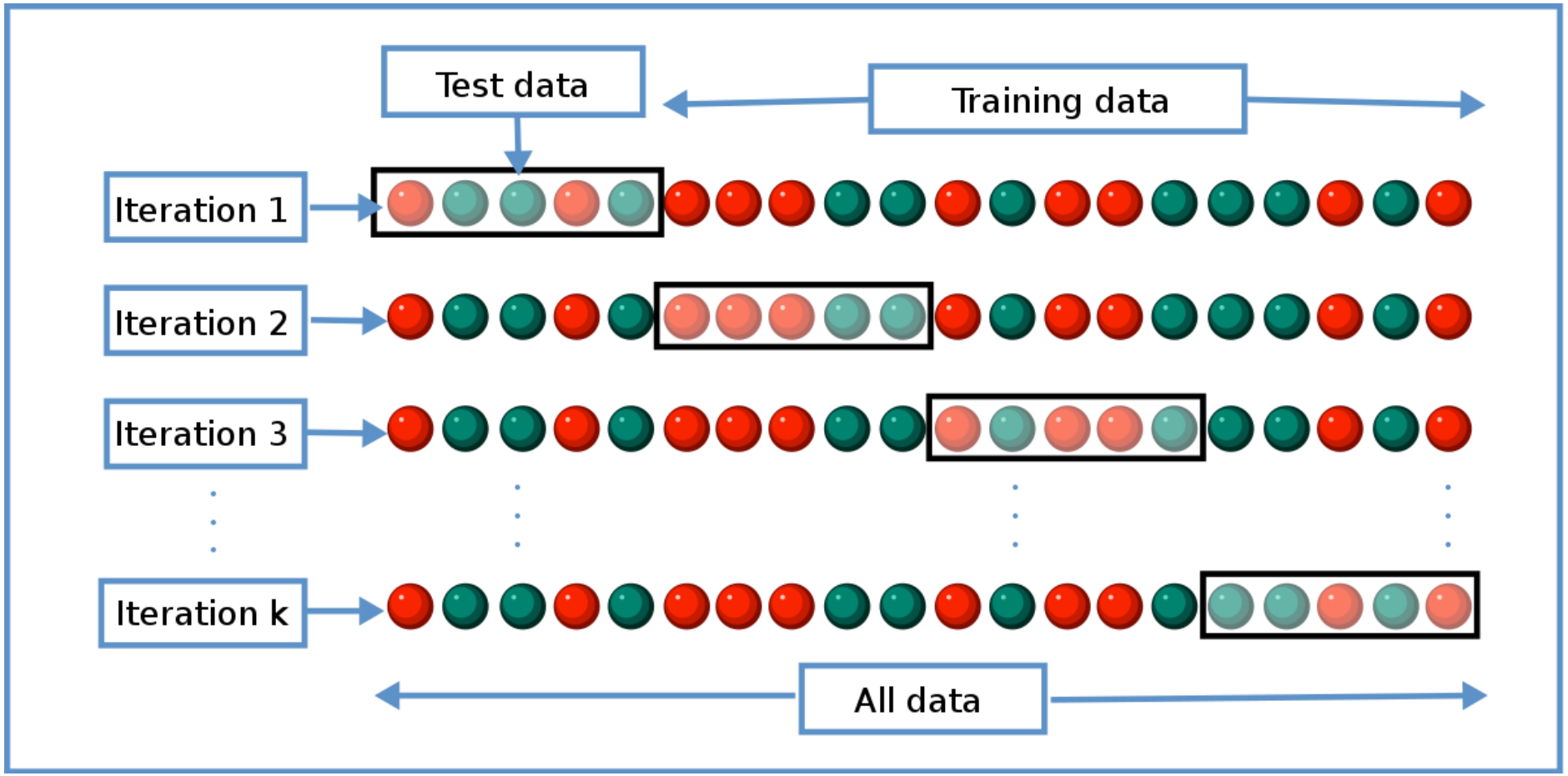
- Validación cruzada es el método más utilizado para entrenar y evaluar un modelo utilizando todos los datos disponibles.

- La idea de validación cruzada es muy simple:
  - Divide los datos en dos conjuntos, entrenamiento y prueba.
  - Entrena el modelo y evalúalo con esta partición.
  - Repite lo mismo varias veces.
  - Después de varias repeticiones, calcula el promedio de los resultados (promedio de las matrices de confusión calculadas en cada paso), así como la varianza.

Validación cruzada

- Existen muchas formas de validación cruzada. La más común se conoce como validación cruzada en k-pliegues (k-fold cross validation):
  - Divide aleatoriamente el conjunto de datos en k particiones o pliegues.
  - Por cada partición i:
    - Entrena con el resto de particiones que no sean la partición i.
    - Evalúa el modelo con la partición i.
  - Calcula el rendimiento promedio obtenido en las k evaluaciones.
  - Entrena el modelo con todos los datos.

Validación cruzada



Validación cruzada

- Si también se requiere calcular los hiperparámetros del modelo, es necesario realizar una validación cruzada anidada.

- Divide aleatoriamente los datos en k_1 particiones.
- Por cada partición i:
  - Divide aleatoriamente en k_2 subparticiones el conjunto de datos formado con todas las k_1 particiones originales excepto la i.
  - Por cada subpartición j:
    - Para cada configuración de hiperparámetros a probar:
      - Entrena el modelo utilizando las subparticiones que no sean la j.
      - Evalúa el modelo con el conjunto de validación obtenido con la subpartición j.
  - Selecciona la configuración de hiperparámetros con la se obtuvieron los mejores resultados de acuerdo al promedio obtenido con las k_2 subparticiones.
  - Entrena el modelo utilizando los hiperparámetros encontrados y los datos de todas las particiones excepto la i.
- Calcula el rendimiento promedio del modelo con las k_1 evaluaciones realizadas.
- Selecciona la configuración de hiperparámetros que haya dado el mejor resultado en las k_1 evaluaciones.
- Entrena el modelo con todos los datos y la configuración de hiperparámetros seleccionada.

Bibliografía

- **An introduction to statistical learning: with applications in** R. Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2da edición, 2015, Springer. Capítulo 4, páginas: 128-167.
- **The elements of statistical learning: data mining, inference, and prediction.** Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2da edición, 2009, Springer. Capítulo 4, páginas: 101-135.
- **Artificial intelligence: a modern approach.** Stuart J. Russell and Peter Norvig. 3ra edición, 2015, Person. Capítulo 18, páginas: 695-757.
- **Artificial intelligence with Python.** Alberto Artasanchez and Prateek Joshi. 2da edición, 2020, Pack. Capítulo 5.