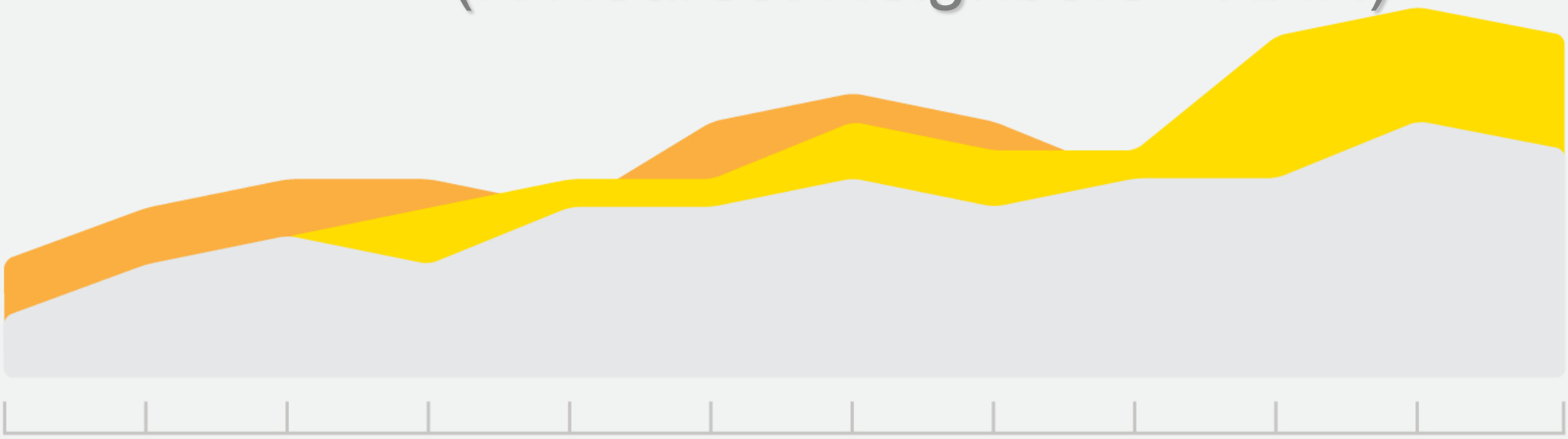
oldemar rodríguez
CONSULTOR en MINERIA DE DATOS

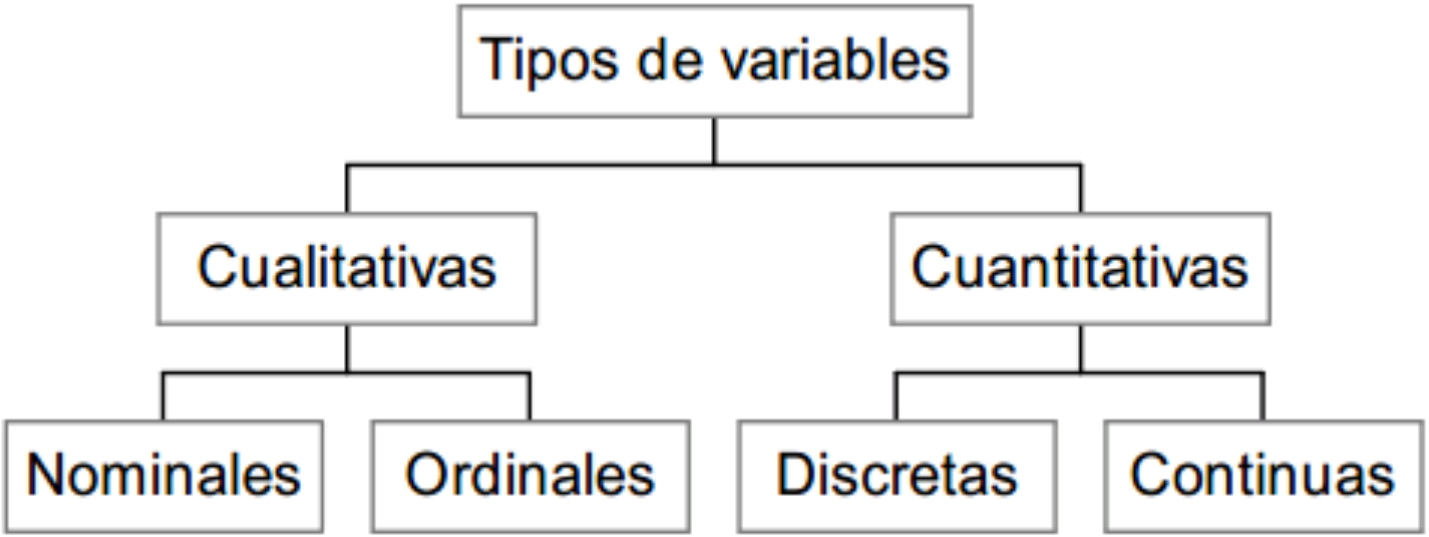
Aprendizaje Supervisado

K vecinos más cercanos

(K Nearest Neighbors - KNN)



Tipos de Variables



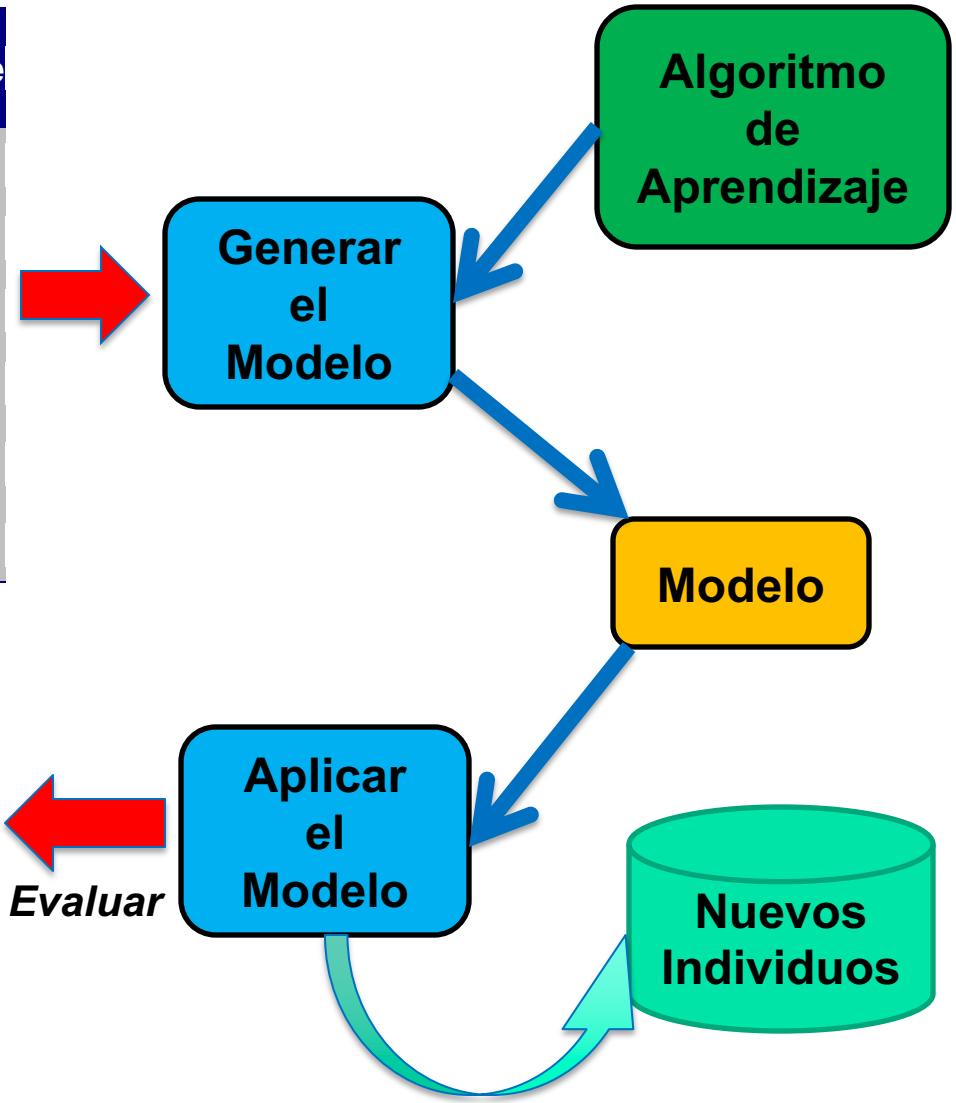
Modelo general de los métodos de Clasificación

Id	Reembolso	Estado Civil	Ingresos Anuales	Fraude
1	Sí	Soltero	125K	No
2	No	Casado	100K	No
3	No	Soltero	70K	No
4	Sí	Casado	120K	No
5	No	Divorciado	95K	Sí
6	No	Casado	60K	No

Tabla de Aprendizaje

Id	Reembolso	Estado Civil	Ingresos Anuales	Fraude
7	No	Soltero	80K	No
8	Si	Casado	100K	No
9	No	Soltero	70K	No

Tabla de Testing



Predicción (Clasificación): Definición

- Dada una colección de registros (conjunto de entrenamiento) cada registro contiene un conjunto de variables (atributos) denominado x, con un variable (atributo) adicional que es la clase denominada y.

- El objetivo de la ***clasificación*** es encontrar un modelo (una función o algortimo) para predecir la clase a la que pertenecería cada registro, esta asignación una clase se debe hacer con la mayor precisión posible.

- Un conjunto de prueba (tabla de testing) se utiliza para determinar la precisión del modelo. Por lo general, el conjunto de datos dado se divide en dos conjuntos al azar de el de entrenamiento y el de prueba.

Definición de Predicción (Clasificación)

- Dada una base de datos $D = \{t_1, t_2, \dots, t_n\}$ de tuplas o registros (individuos) y un conjunto de clases $C = \{C_1, C_2, \dots, C_m\}$, el **problema de la clasificación** es encontrar una función $f: D \to C$ tal que cada t_i es asignada una clase C_j.

- $f: D \to C$ podría ser una Red Neuronal, un Árbol de Decisión, un modelo basado en Análisis Discriminante, o una Red Beyesiana.

Ejemplo 1: Credit-Scoring

Muestra5000V2.csv

Cargar | Transformar | Configuraciones

Variables	Tipo	Activa
MontoCredito	Numérico	✓
IngresoNeto	Categórico	✓
CoefCreditoAvaluo	Categórico	✓
MontoCuota	Categórico	✓
GradoAcademico	Categórico	✓
BuenPagador	Categórico	✓

Aplicar

```
datos <<- datos.originales
datos[, 'IngresoNeto'] <<- as.factor(datos[, 'IngresoNeto'])
datos[, 'CoefCreditoAvaluo'] <<- as.factor(datos[, 'CoefCreditoAvaluo'])

datos <<- subset(datos, select = -c())
```

Datos

Descargar

Search:

ID	MontoCredito	IngresoNeto	CoefCreditoAvaluo	MontoCuota	GradoAcademico	BuenPagador
1	14327	1	1	MuyBajo	Bachiller	Si
2	111404	1	1	MuyBajo	Bachiller	Si
3	21128	1	1	MuyBajo	Bachiller	Si
4	15426	2	1	MuyBajo	Bachiller	Si
5	10351	1	1	MuyBajo	Bachiller	Si
6	27060	1	1	MuyBajo	Bachiller	Si
7	243369	1	1	MuyBajo	Bachiller	Si
8	16300	2	1	MuyBajo	Bachiller	Si
9	18319	2	1	MuyBajo	Bachiller	Si
10	107037	2	1	MuyBajo	Bachiller	Si
	Numérico	Categórico	Categórico	Categórico	Categórico	Categórico

Showing 1 to 10 of 5,000 entries

Previous 1 2 3 4 5 … 500 Next

Descripción de Variables

MontoCredito

Numérica

MontoCuota

1=Muy Bajo
2=Bajo
3=Medio
4=Alto

IngresoNeto

1=Muy Bajo
2=Bajo
3=Medio
4=Alto

GradoAcademico

1=Bachiller
2=Licenciatura
3=Maestría
4=Doctorado

CoeficienteCreditoAvaluo

1=Muy Bajo
2=Bajo
3=Medio
4=Alto

BuenPagador

1=NO
2=Si

Ejemplo: Créditos en un Banco

Tabla de Aprendizaje

Variable a Predecir

OLDEMARRR.DMEx...ditoViviendaPeq

Id	MontoCredito	IngresoNeto	CoeficienteCre...	MontoCuota	GradoAcademico	BuenPagador
1	2	4	3	1	4	1
2	2	3	2	1	4	1
3	4	1	1	4	2	2
4	1	4	3	1	4	1
5	3	3	1	3	2	2
6	3	4	3	1	4	1
7	4	2	1	3	2	2
8	4	1	3	3	2	2
9	3	4	3	1	3	1
10	1	3	2	2	4	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Con la Tabla de Aprendizaje se entrena (aprende) el modelo matemático de predicción, es decir, a partir de esta tabla se calcula la función ***f*** de la definición anterior.

Ejemplo: Créditos en un Banco

Tabla de Testing

Variable a Predecir

OLDEMARRR.DME...iviendaPeqPRED | OLDEMARRR.DMEx...ditoViviendaPeq

	Id	MontoCredito	IngresoNeto	CoeficienteCre...	MontoCuota	GradoAcademico	BuenPagador
▶	11	3	3	3	3	1	2
	12	2	2	2	2	1	1
	13	2	2	3	2	1	1
	14	1	3	4	3	2	2
	15	1	2	4	2	1	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- Con la Tabla de Testing se valida el modelo matemático de predicción, es decir, se verifica que los resultados en individuos que no participaron en la construcción del modelo es bueno o aceptable.
- Algunas veces, sobre todo cuando hay pocos datos, se utiliza la Tabla de Aprendizaje también como de Tabla Testing.

Ejemplo: Créditos en un Banco

Nuevos Individuos

Variable a Predecir

OLDEMARRR.DMEx ...editoViviendaNI

	Id	MontoCredito	IngresoNeto	CoeficienteCre...	MontoCuota	GradoAcademico	BuenPagador
	100	4	4	2	2	3	?
	101	1	4	3	2	4	?
	102	3	2	3	4	2	?
▶*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

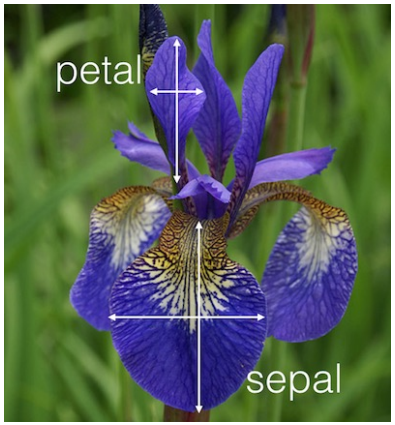
Con la Tabla de Nuevos Individuos se predice si estos serán o no buenos pagadores.

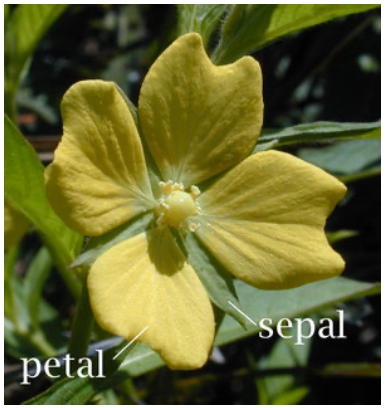
Ejemplo 2: IRIS.CSV (Fisher)

Ejemplo con la tabla de datos IRIS
IRIS Información de variables:

1.sepal largo en cm
2.sepal ancho en cm
3.petal largo en cm
4.petal ancho en cm
5.clase:

- Iris Setosa
- Iris Versicolor
- Iris Virginica

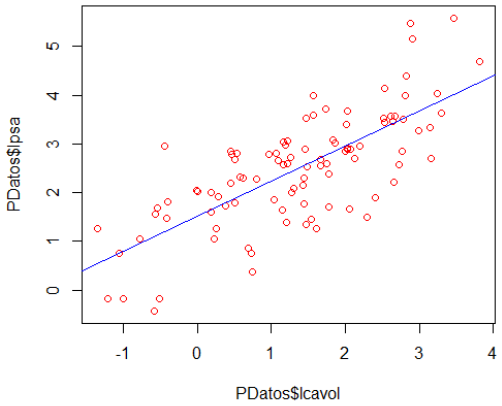




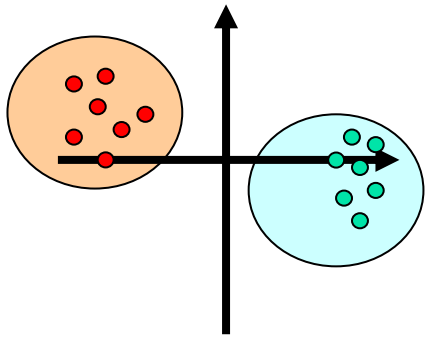
	A	B	C	D	E
1	s.largo	s.ancho	p.largo	p.ancho	tipo
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3.0	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa
6	5.0	3.6	1.4	0.2	setosa
7	5.4	3.9	1.7	0.4	setosa
8	4.6	3.4	1.4	0.3	setosa
9	5.0	3.4	1.5	0.2	setosa
10	4.4	2.9	1.4	0.2	setosa
11	4.9	3.1	1.5	0.1	setosa
12	5.4	3.7	1.5	0.2	setosa
13	4.8	3.4	1.6	0.2	setosa
14	4.8	3.0	1.4	0.1	setosa
15	4.3	3.0	1.1	0.1	setosa
16	5.8	4.0	1.2	0.2	setosa
17	5.7	4.4	1.5	0.4	setosa
18	5.4	3.9	1.3	0.4	setosa
19	5.1	3.5	1.4	0.3	setosa
20	5.7	3.8	1.7	0.3	setosa
21	5.1	3.8	1.5	0.3	setosa
22	5.4	3.4	1.7	0.2	setosa
23	5.1	3.7	1.5	0.4	setosa
24	4.6	3.6	1.0	0.2	setosa
25	....	...	...	...	....

Regresión vs Clasificación

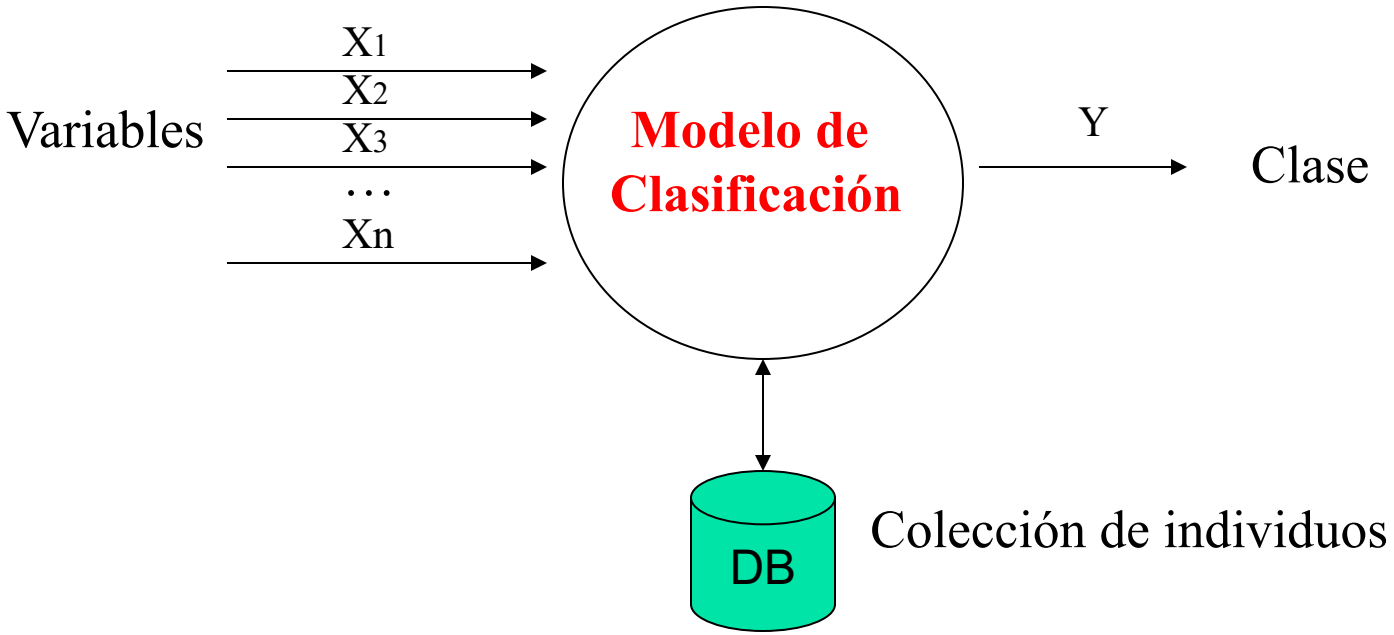
- Regresión:
  - La variable a predecir es cuantitativa
  - Por ejemplo predecir el salario de una persona



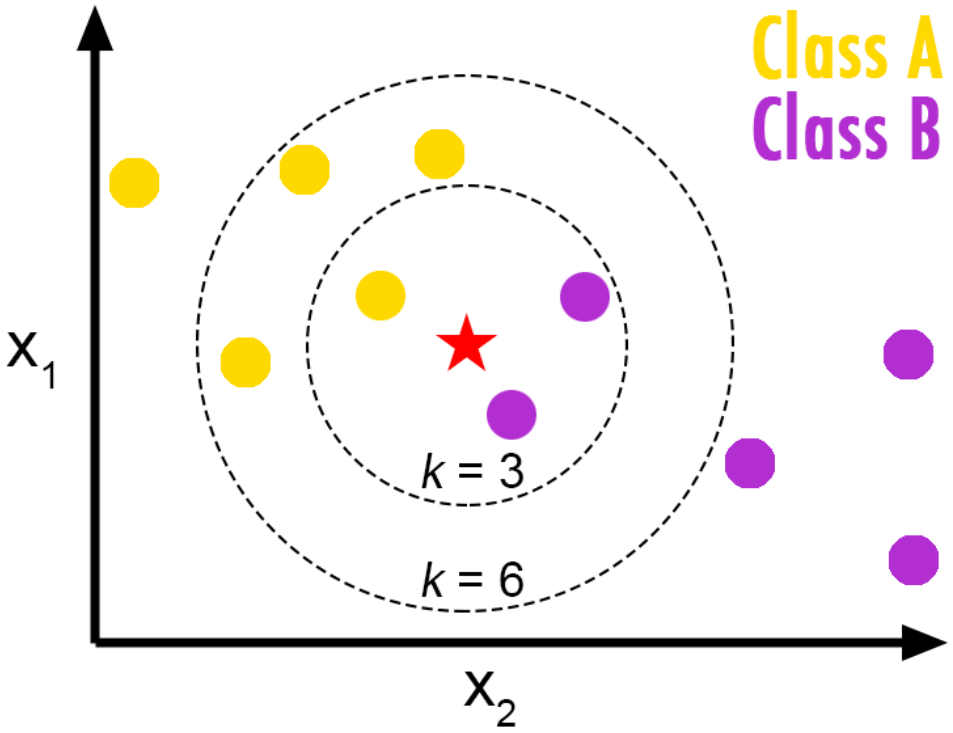
- Clasificación
  - La variable a predecir es cualitativa
  - Por ejemplo predecir si una transacción es fraude o no



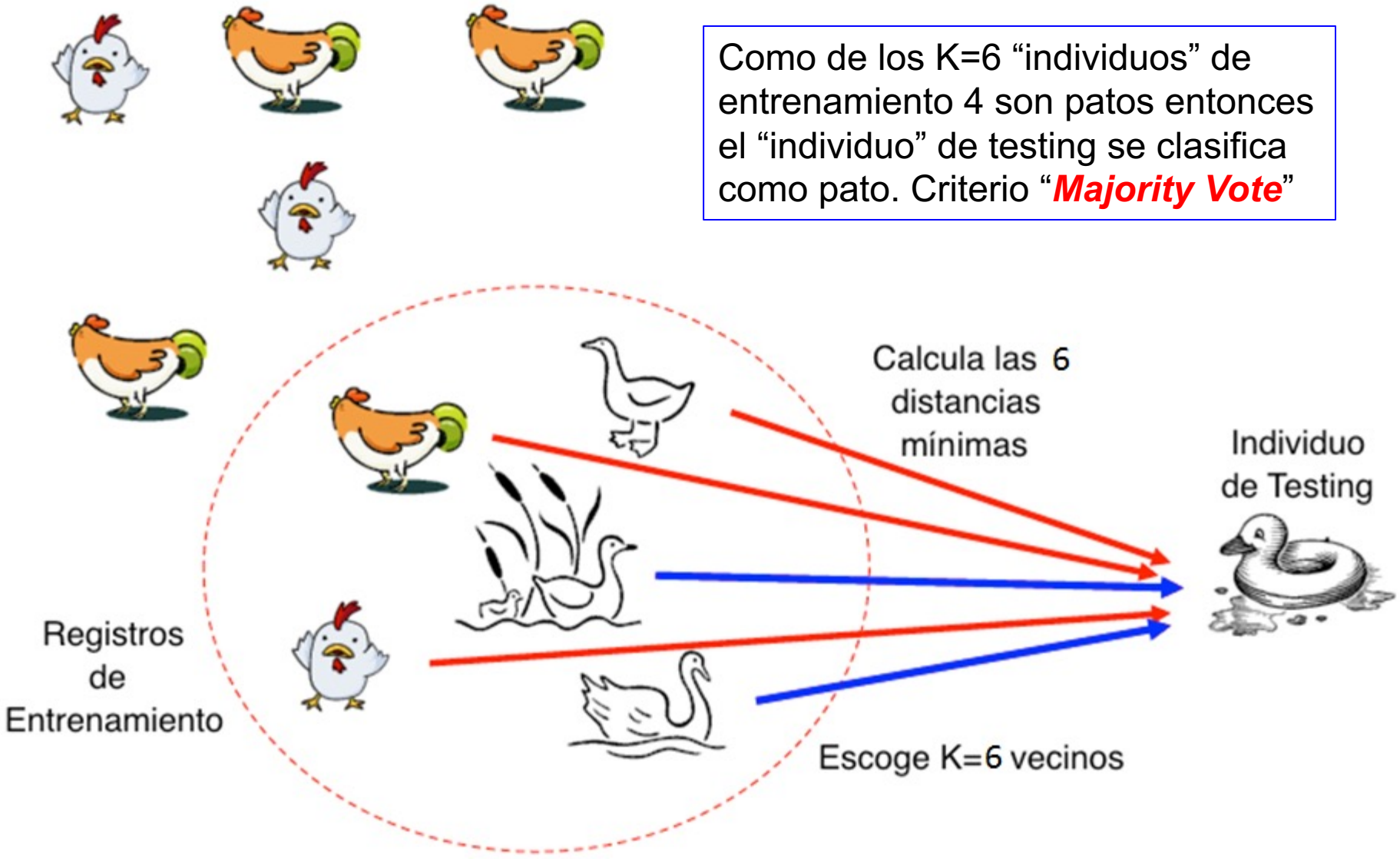
Modelos Predictivos

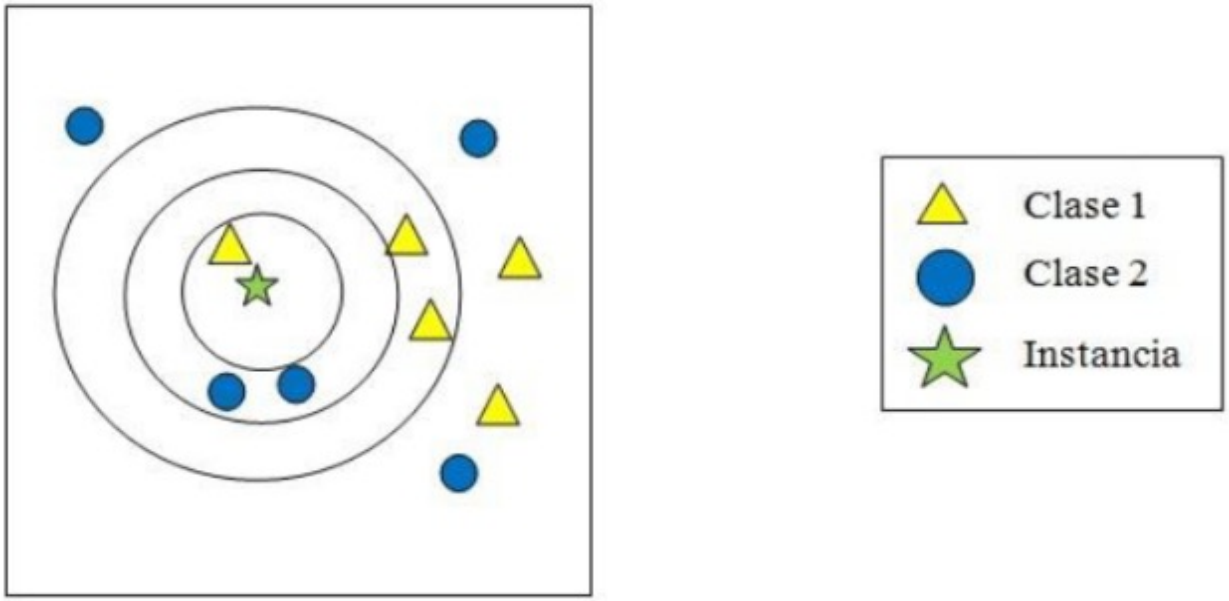


Método de los
K vecinos más cercanos
(K Nearest Neighbors - KNN)



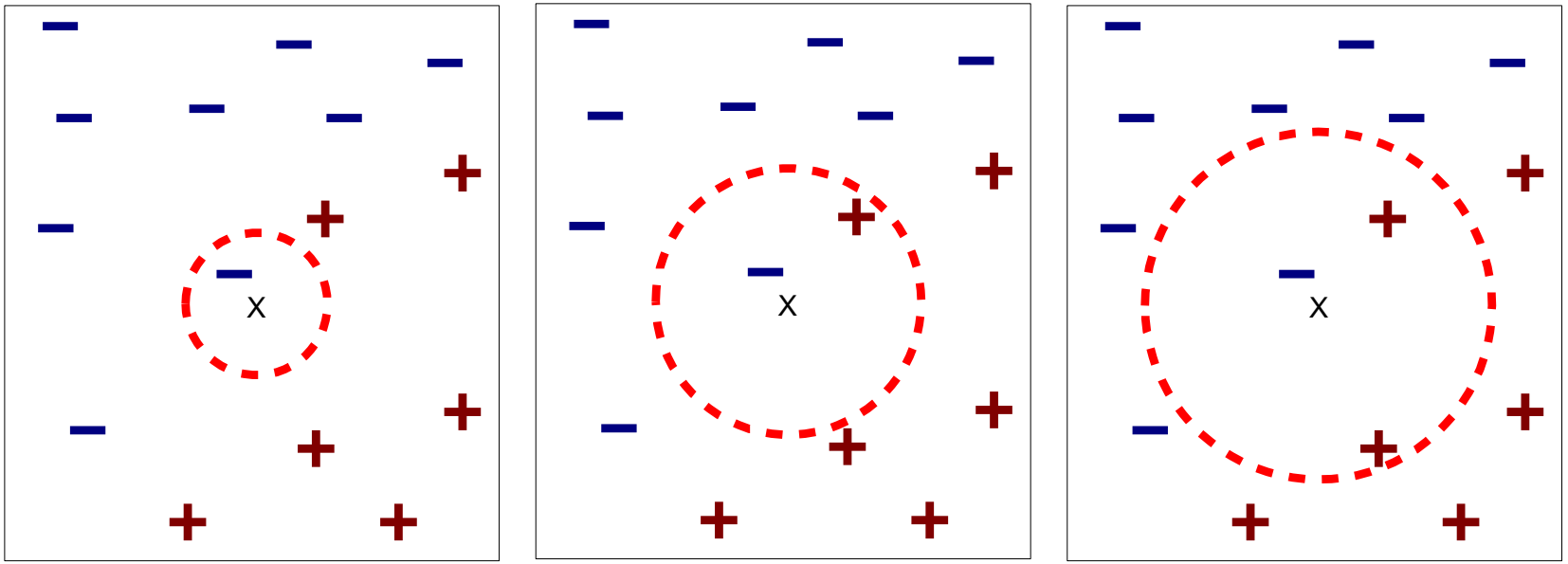
Como de los K=6 “individuos” de entrenamiento 4 son patos entonces el “individuo” de testing se clasifica como pato. Criterio “***Majority Vote***”





Para K=1 (círculo más pequeño), la clase de la nueva instancia sería la Clase 1, ya que es la clase de su vecino más cercano, mientras que para K=3 la clase de la nueva instancia sería la Clase 2 pues habrían dos vecinos de la Clase 2 y solo 1 de la Clase 1.

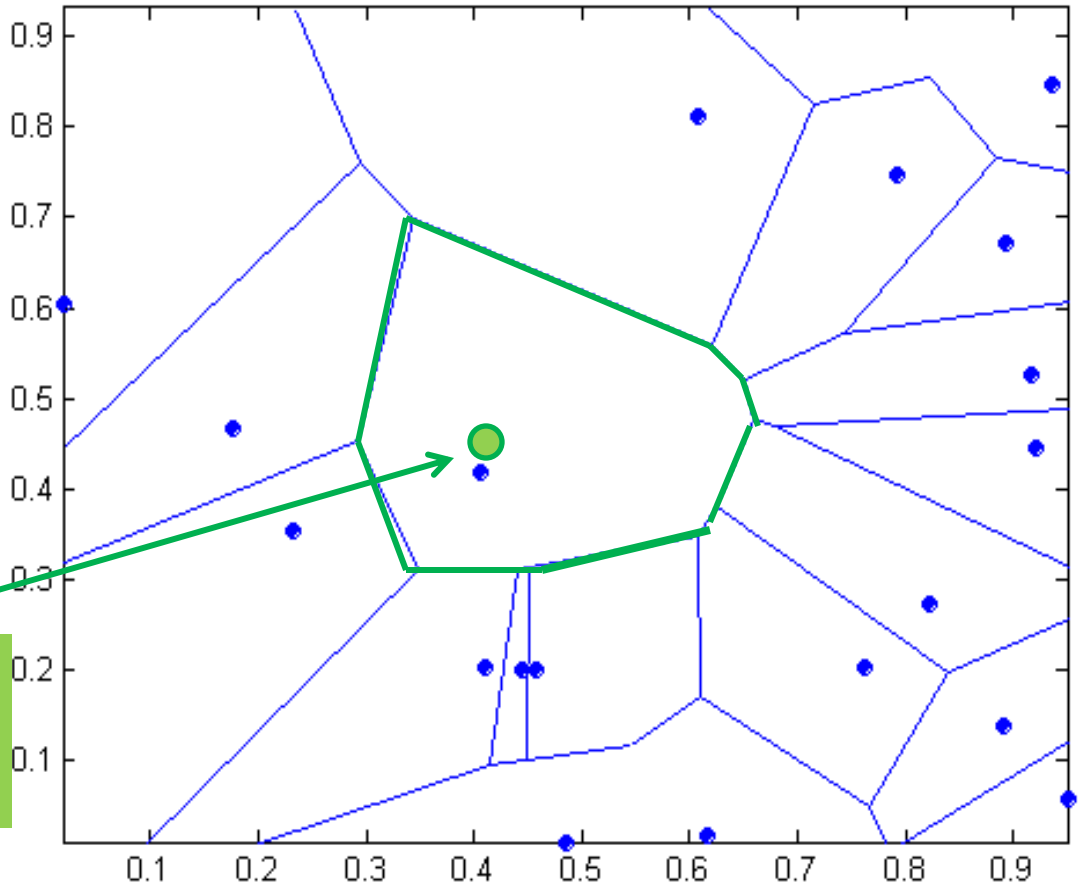
¿Cómo escoger K?



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

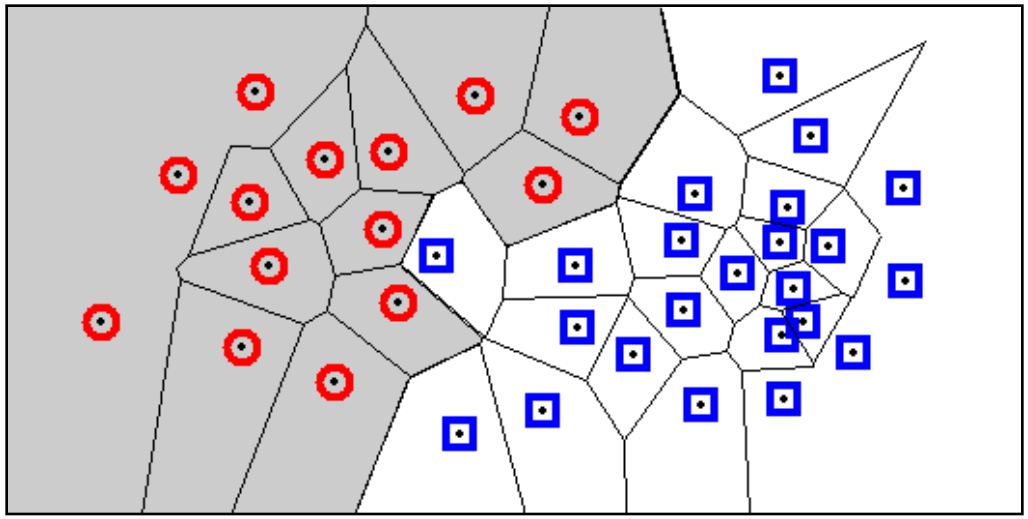
1 Vecino más cercano

El ***Diagrama de Voronoi*** define las fronteras de la clasificación

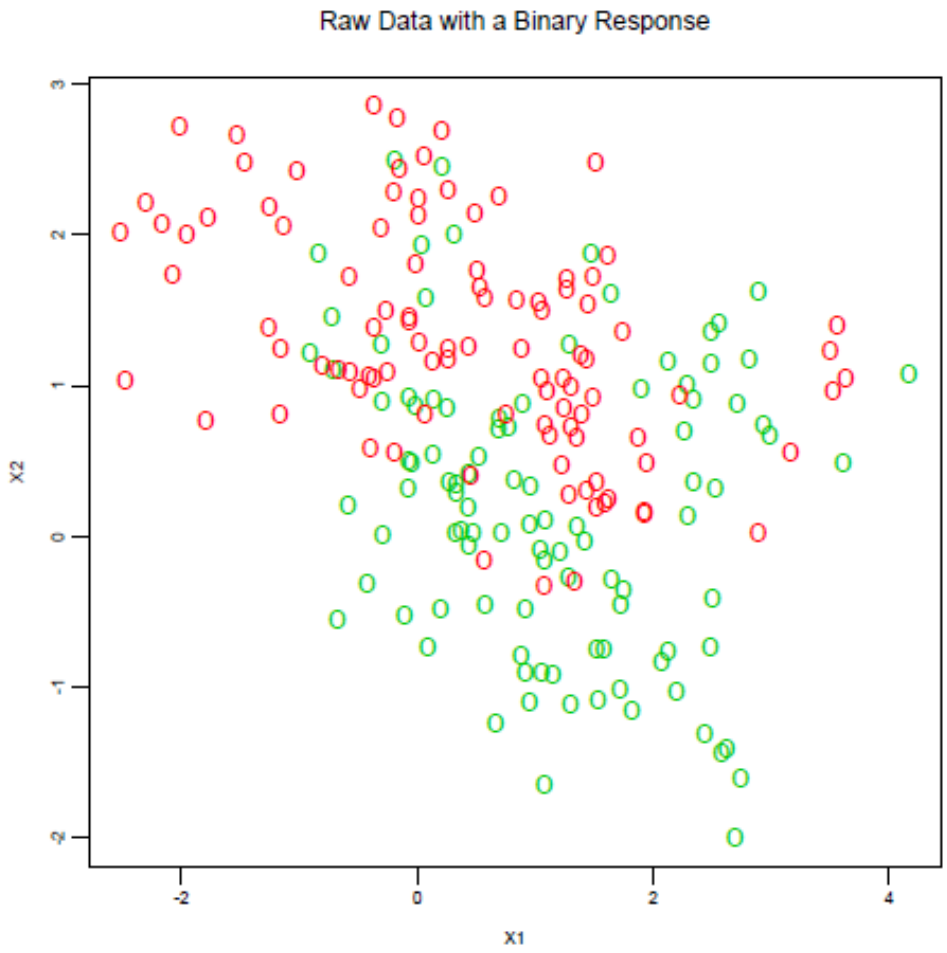


Esta es el área tomada por el punto verde

Diagrama de Voronoi Regiones Decisión

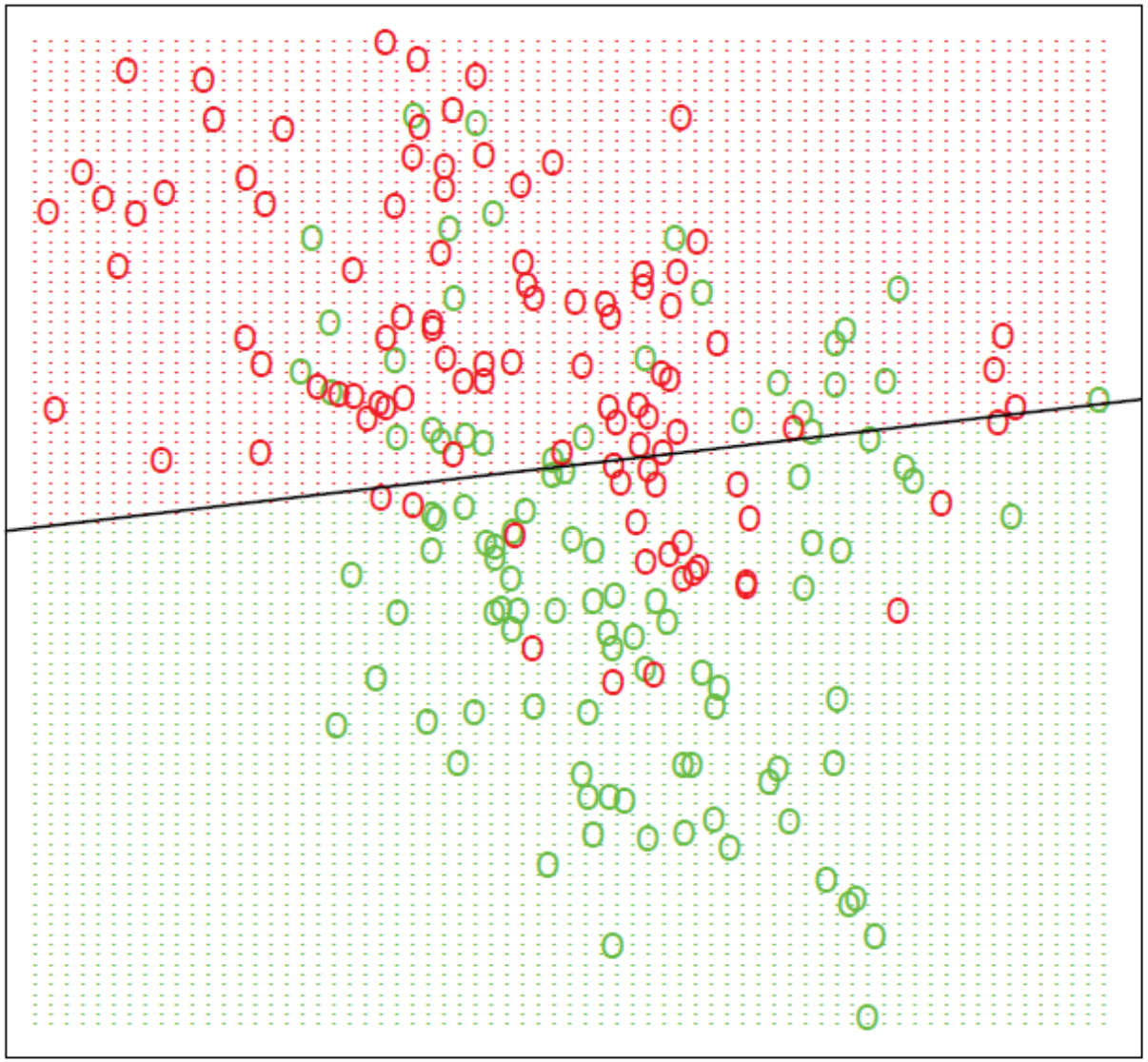


El diagrama de Voronoi divide el espacio en celdas

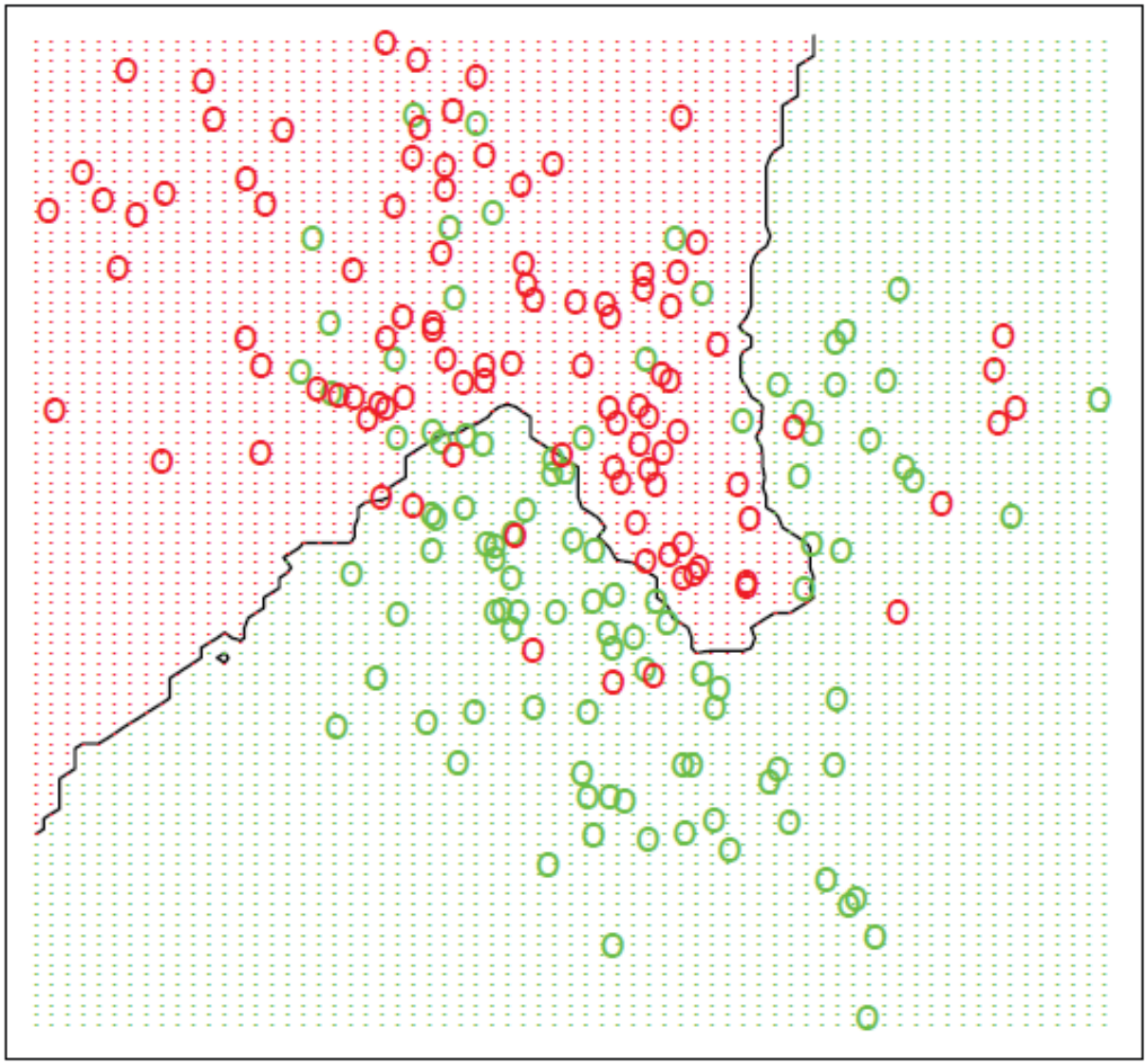


200 points generated in $\mathbb{R}^2$ from an unknown distribution; 100 in each of two classes $\mathcal{G} = \{\texttt{GREEN}, \texttt{RED}\}$. Can we build a rule to predict the color of future points?

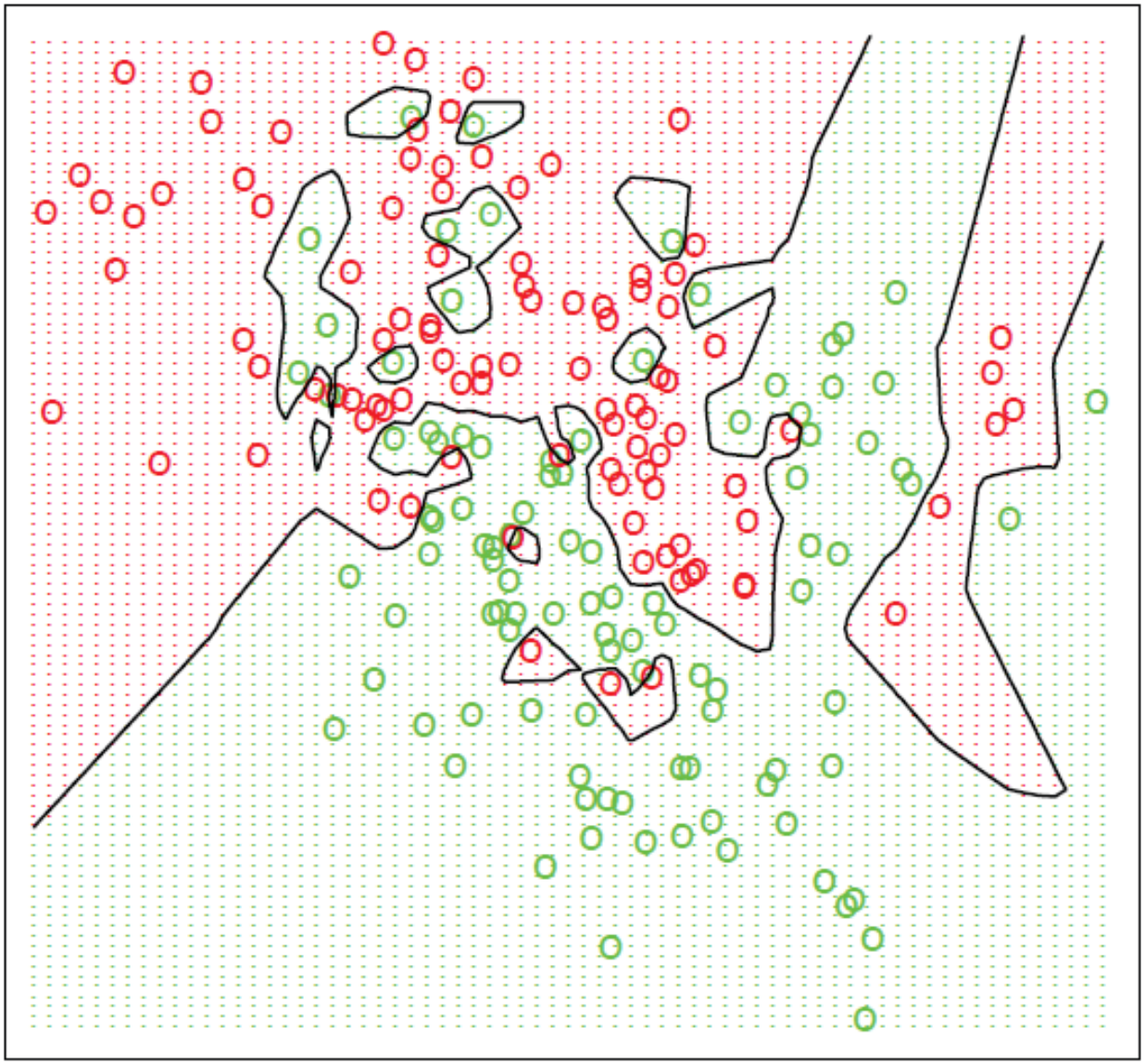
Linear Regression of 0/1 Response



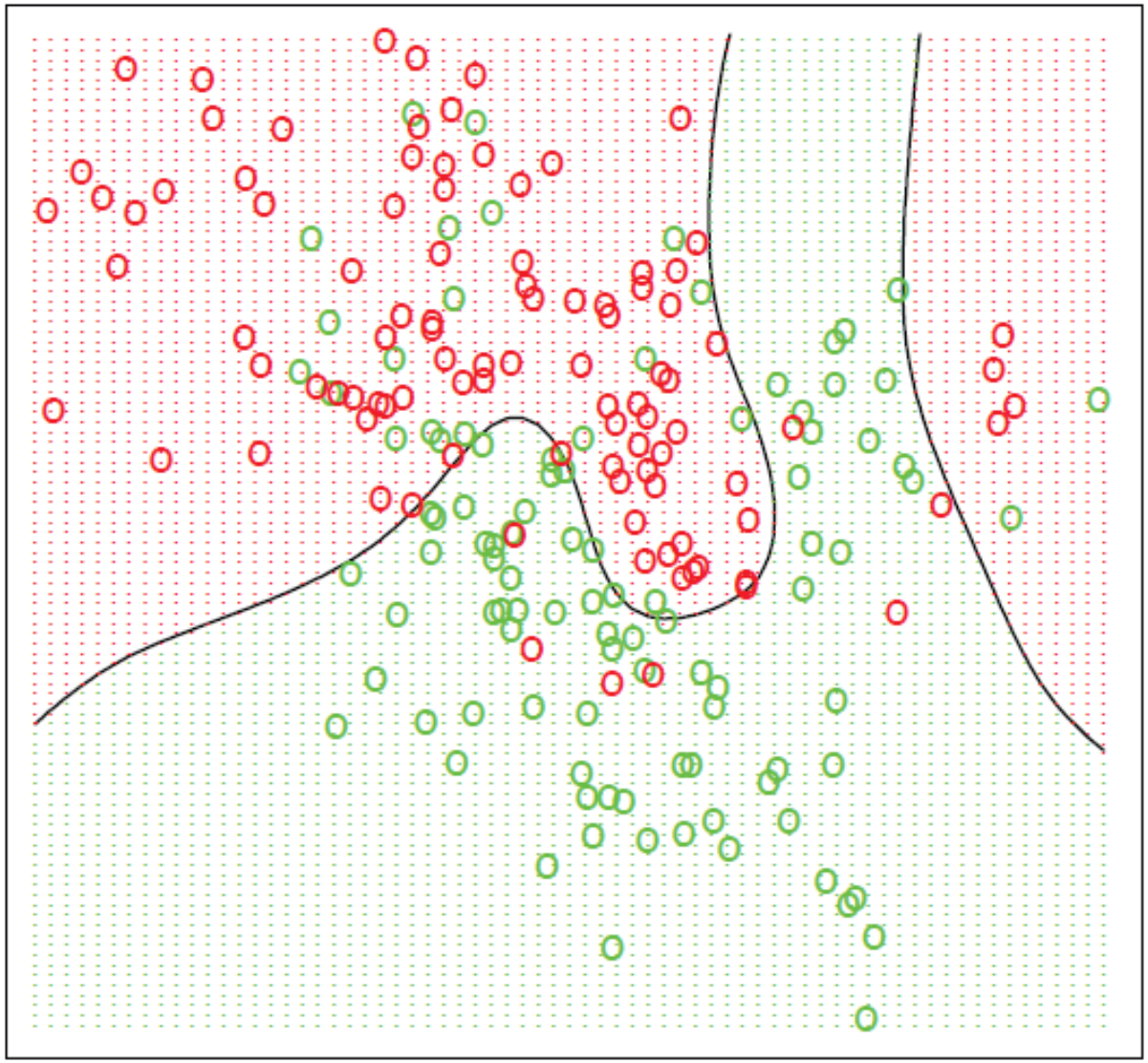
15-Nearest Neighbor Classifier



1-Nearest Neighbor Classifier



Bayes Optimal Classifier



Decisiones Importantes
K vecinos más cercanos

- Medida distancia a utilizar
- Valor de *k* (generalmente impar)
- Mecanismo de votación
- Indexación memoria

K vecinos más cercanos

- Ventajas
  - La arquitectura no paramétrica
  - Método simple
  - Potente
  - No requiere mucho tiempo de entrenamiento
- Desventajas
  - Memoria intensiva
  - Clasificación / estimación es lenta

Parámetros de entrenamiento y ajustes típicos

- ¿Cómo escoger el número de vecinos más cercanos?
  - El número de vecinos más cercanos (K) se puede calcular usando ***Validación Cruzada*** sobre un número de ajuste K.
  - Cuando K = 1 es un buen modelo de base de referencia contra el cual comparar.
  - Una buena regla para escoger K es que debe ser menor que la raíz cuadrada del número total de individuos en la tabla de entrenamiento.

¿Cómo evaluar la calidad del Modelo Predictivo?

Matriz de confusión (Matriz de Error)

- La ***Matriz de Confusión*** contiene información acerca de las predicciones realizadas por un ***Método o Sistema de Clasificación***, comparando para el conjunto de individuos en de la tabla de aprendizaje o de testing, la predicción dada versus la clase a la que estos realmente pertenecen.
- La siguiente tabla muestra la matriz de confusión para un clasificador de dos clases:

		Predicción	
		Negativo	Positivo
Valor Real	Negativo	VN	FP
	Positivo	FN	VP

Ejemplo: Matriz de confusión

		Predicción	
		Mal Pagador	Buen Pagador
Valor Real	Mal Pagador	800	200
	Buen Pagador	500	1500

- 800 predicciones de Mal Pagador fueron realizadas correctamente, para un 80%, mientras que 200 no, para un 20%.
- 1500 predicciones de Buen Pagador fueron realizadas correctamente, para un 75%, mientras que 500 no (para un 25%).
- En general 2300 de 3000 predicciones fueron correctas para un 76,6% de efectividad en las predicciones. ***Cuidado***, este dato es a veces engañoso y debe ser siempre analizado en la relación a la dimensión de las clases.

Matriz de confusión

		Predicción	
		Negativo	Positivo
Valor Real	Negativo	VN	FP
	Positivo	FN	VP

- La Precisión Global ***P*** (Exactitud) de un modelo de predicción es la proporción del número total de predicciones que son correctas respecto al total. Se determina utilizando la ecuación:

$$P = (VN+VP)/(VN+FP+FN+VP)$$

- ***Cuidado***, este índice es a veces engañoso y debe ser siempre analizado en la relación a la dimensión de las clases.

Ejemplo: Matriz de confusión

		Predicción	
		Fraude	No Fraude
Valor Real	Fraude	0	8
	No Fraude	3	989

- ***Cuidado***, este índice es a veces engañoso y debe ser siempre analizado en la relación a la dimensión de las clases.

- En la Matriz de Confusión anterior la Precisión ***P*** es del 98,9%, sin embargo, el modelo no detectó ningún fraude.

Matriz de confusión

		Predicción	
		Negativo	Positivo
Valor Real	Negativo	VN	FP
	Positivo	FN	VP

- La Precisión Positiva (Sensibilidad) (***PP***) (Porcentaje de Verdaderos Positivos) es la proporción de casos positivos que fueron identificados correctamente, tal como se calcula usando la ecuación:

$$PP = VP/(FN+VP)$$

- En el ejemplo anterior Precisión Positiva ***PP*** es del 99,6% .

Matriz de confusión

		Predicción	
		Negativo	Positivo
Valor Real	Negativo	VN	FP
	Positivo	FN	VP

- La Precisión Negativa (Especifidad) (***PN***) es la proporción de casos negativos que fueron identificados correctamente, tal como se calcula usando la ecuación:

$$PN = VN/(VN+FP)$$

- En el ejemplo anterior Precisión Negativa ***PN*** es del 0% .

Matriz de confusión

		Predicción	
		Negativo	Positivo
Valor Real	Negativo	VN	FP
	Positivo	FN	VP

- Falsos Positivos (***PFP***) es la proporción de casos negativos que fueron clasificados incorrectamente como positivos, tal como se calcula utilizando la ecuación:

$$PFP = FP/(VN+FP)$$

- Falsos Negativos (***PFN***) es la proporción de casos positivos que fueron clasificados incorrectamente como negativos, tal como se calcula utilizando la ecuación:

$$PFN = FN/(FN+VP)$$

Matriz de confusión

		Predicción	
		Negativo	Positivo
Valor Real	Negativo	VN	FP
	Positivo	FN	VP

- Asertividad Positiva (***AP***) indica la proporción de buena predicción para los positivos, tal como se calcula utilizando la ecuación:

$$AP = VP/(FP+VP)$$

- Asertividad Negativa (***AN***) indica la proporción de buena predicción para los negativos, tal como se calcula utilizando la ecuación:

$$AN = VN/(VN+FN)$$

Matriz de confusión para más de 2 clases

- La Matriz de Confusión puede calcularse en general para un problema con ***p*** clases.

- En la matriz ejemplo que aparece a continuación, de 8 alajuelenses reales, el sistema predijo que 3 eran heredianos y de 6 heredianos predijo que 1 era un limonense y 2 eran alajuelenses. A partir de la matriz se puede ver que el sistema tiene problemas distinguiendo entre alajuelenses y heredianos, pero que puede distinguir razonablemente bien entre limonenses y las otras provincias.

		Predicción		
		alajuelense	herediano	limonense
Valor Real	alajuelense	5	3	0
	herediano	2	3	1
	limonense	0	2	11

sklearn.neighbors.KNeighborsClassifier

class `sklearn.neighbors.`**`KNeighborsClassifier`**(*n_neighbors=5*, ***, *weights='uniform'*, *algorithm='auto'*, *leaf_size=30*, *p=2*, *metric='minkowski'*, *metric_params=None*, *n_jobs=None*) [source]

Classifier implementing the k-nearest neighbors vote.

Read more in the User Guide.

Parameters::

n_neighbors : *int, default=5*
Number of neighbors to use by default for `kneighbors` queries.

weights : *{'uniform', 'distance'} or callable, default='uniform'*
Weight function used in prediction. Possible values:

- 'uniform' : uniform weights. All points in each neighborhood are weighted equally.
- 'distance' : weight points by the inverse of their distance. in this case, closer neighbors of a query point will have a greater influence than neighbors which are further away.
- [callable] : a user-defined function which accepts an array of distances, and returns an array of the same shape containing the weights.

algorithm : *{'auto', 'ball_tree', 'kd_tree', 'brute'}, default='auto'*
Algorithm used to compute the nearest neighbors:

- 'ball_tree' will use `BallTree`
- 'kd_tree' will use `KDTree`
- 'brute' will use a brute-force search.

Link a la documentación:

https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

oldemar rodríguez

CONSULTOR en M1NЄR14 DЄ D4T0S

Gracias....