Clasificación usando Vecino más Cercano

Jorge Gallego

Facultad de Economía, Universidad del Rosario

febrero 14 de 2017

Introducción

- Tras la breve introducción a R, volvamos a nuestro tema central
- Recordemos nuestra definición de *machine learning*
- Es el campo interesado en desarrollar algoritmos para transformar los datos en acción inteligente
- Como el spam de gmail, los carros sin conductor, Alexa de Amazon, las recomendaciones en Netflix

Encontrando a Rover

EL TIEMPO

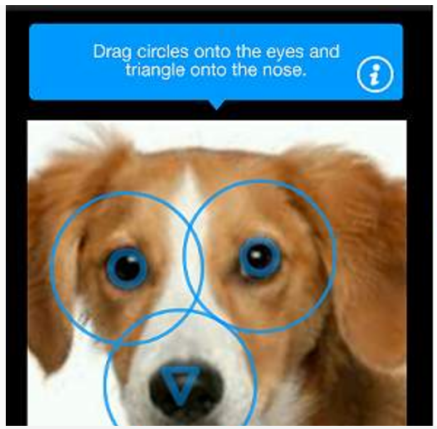
JORGE

TECNÓSFERA NOVEDADES RESEÑAS TUTORIALES VIDEOJUEGOS

La 'app' de reconocimiento facial para que perros perdidos regresen

'Finding Rover' ha ayudado a más de 2.000 mascotas en Estados Unidos a reunirse con sus familias.

Por: ANA MARÍA VELÁSQUEZ DURÁN |
1:04 p.m. | 24 de enero de 2017



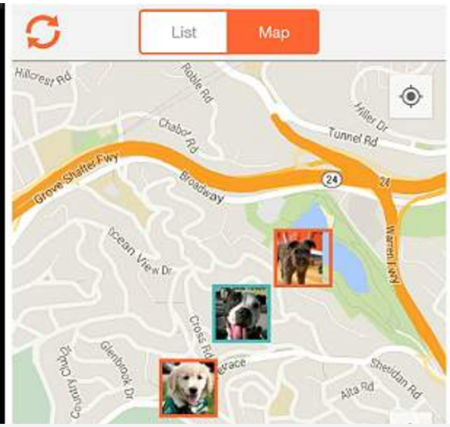


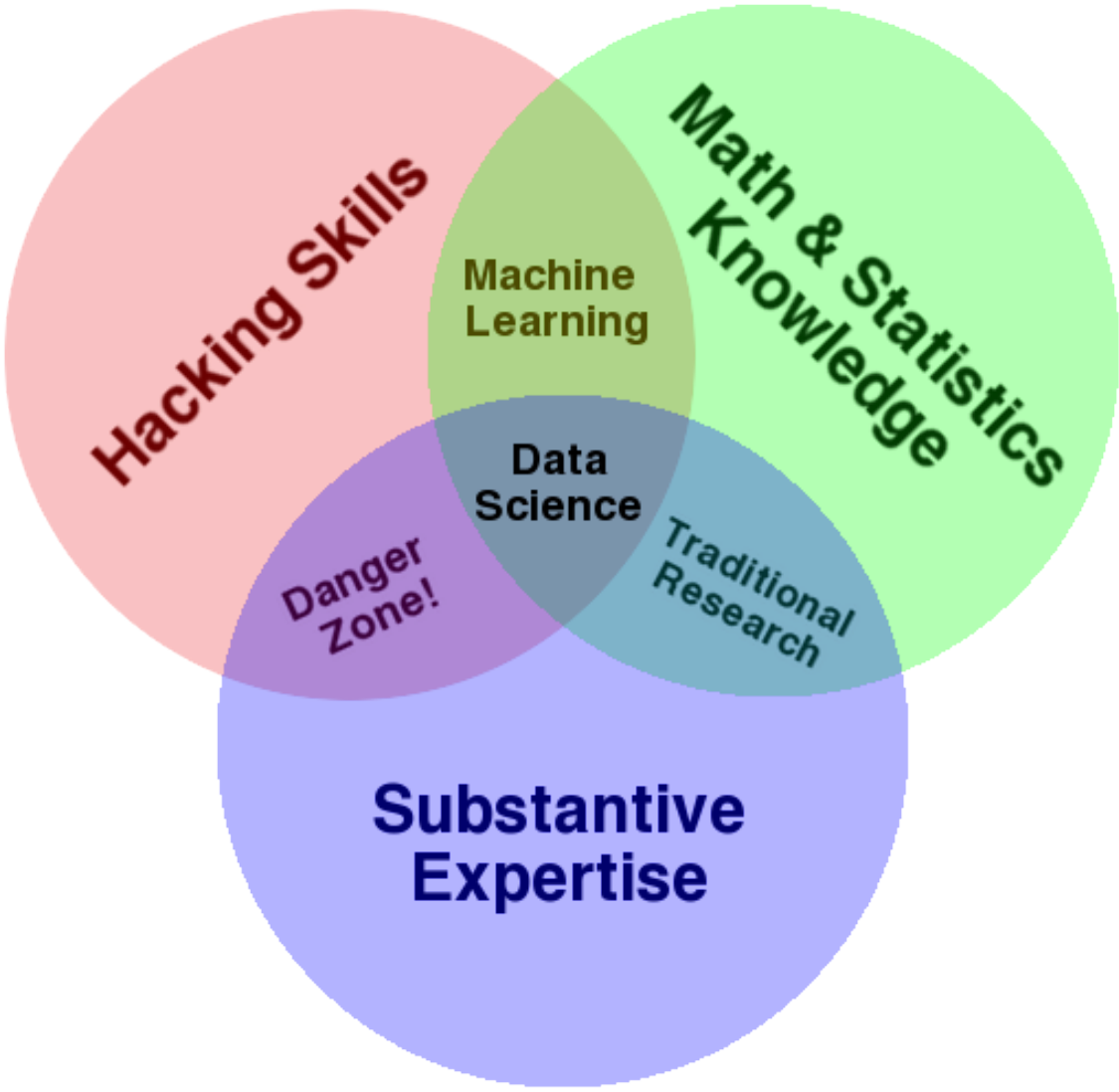
Foto: Archivo particular

Los usuarios deben marcar los ojos y nariz en la foto para que luego el sistema se encargu...

3252
COMPARTIDOS

Dejó a un lado su trabajo durante cuatro días, recorrió su barrio por horas, habló con todo aquel que se le cruzara. El día en que Steve Cox perdió a 'Roxy', una perra de 10 años de raza Shiba Inu, su vida prácticamente se detuvo.

Diagrama de Conway



Introducción

- Para hacer *machine learning* combinamos tres cosas:

  1. Datos disponibles
  2. Poder computacional
  3. Métodos estadísticos

- El rápido desarrollo de estos tres campos ha promovido la expansión del *machine learning*

- En adelante estudiaremos los principales algoritmos desarrollados hasta el momento

Algoritmos de *Machine Learning*

Model	**Learning task**
Supervised Learning Algorithms	
Nearest Neighbor	Classification
Naive Bayes	Classification
Decision Trees	Classification
Classification Rule Learners	Classification
Linear Regression	Numeric prediction
Regression Trees	Numeric prediction
Model Trees	Numeric prediction
Neural Networks	Dual use
Support Vector Machines	Dual use
Unsupervised Learning Algorithms	
Association Rules	Pattern detection
k-means clustering	Clustering
Meta-Learning Algorithms	
Bagging	Dual use
Boosting	Dual use
Random Forests	Dual use

Algoritmos de *Machine Learning*

- **Modelos predictivos**: predicción de un valor usando otros valores de los datos
- Un modelo de aprendizaje busca descubrir la relación entre una característica (outcome) de interés y las demás
- No siempre tienen que ser predicciones del futuro
- **Aprendizaje supervisado**: optimizar función para encontrar combinación de características que resulten en el outcome

Algoritmos de *Machine Learning*

- **Clasificación**: aprendizaje supervisado para predecir a qué categoría pertenece un caso
  - Correo spam
  - Célula cancerosa
  - Equipo deportivo que ganará o perderá
  - Aplicante que hará default crediticio
- El outcome de interés a predecir es una variable categórica llamada clase.
- Las categorías posibles son los niveles

Algoritmos de *Machine Learning*

- El aprendizaje supervisado también puede usarse para hacer predicciones numéricas
- Por ejemplo ingreso, valores de laboratorio, puntajes o conteos.
- **Modelo descriptivo**: busca resumir los datos de formas útiles y novedosas
- En contraste con un modelo predictivo, ninguna característica es más importante que las demás
- El proceso de entrenar un modelo descriptivo se conoce como **aprendizaje no-supervisado**

Algoritmos de *Machine Learning*

- Los modelos de aprendizaje no supervisado suelen usarse en minería de datos
- **Descubrimiento de patrones**: se usa para identificar asociaciones útiles dentro de los datos
- Ejemplo: análisis de canastas de mercado. Identifican qué artículos tienden a ser comprados al tiempo
- Patrones en comportamiento fraudulento, defectos genéticos o hot spots de actividad criminal

Algoritmos de *Machine Learning*

- **Clustering**: División de una base de datos en grupos homogéneos.
- **Análisis de segmentación**: clustering que identifica grupos de individuos con comportamiento o info demográfica similar
- Útil para campañas publicitarias. Incluso en política
- La máquina construye clusters. El humano interpreta
- **Meta-aprendizaje**: algoritmos para a aprender a aprender mejor

Clasificación usando Vecino más Cercano

- Para hablar de clasificación partamos de un principio básico
- Las cosas que son parecidas tienen propiedades parecidas
- "Blanco es, gallina lo pone, frito se come"
- Los algoritmos de *machine learning* usan este principio para clasificar datos
- En una misma categoría son clasificados elementos similares, o *vecinos más cercanos*

Algoritmos kNN

- El algortimo de vecino más cercano más popular es el kNN (*k nearest neighbor*)
- Tiene tres grandes ventajas:
  - Simple y efectivo
  - No hace supuestos sobre la distribución de los datos
  - Fase rápida de entrenamiento
- Pero también tiene desventajas:
  - No produce un modelo, luego es difícil establecer la relación entre variables
  - Requiere la selección apropiada de k
  - Fase de clasificación lenta
  - Características nominales y datos ausentes requieren procesamiento adicional

Clasificación usando Vecino más Cercano

- El nombre obedece a que los casos no clasificados se categorizan usando los k vecinos más cercanos
- Tras elegir k, el algoritmo requiere el entrenamiento de un conjunto de datos en los que ya están clasificados
- Luego, para cada caso no clasificado se buscan los k vecinos más cercanos
- El caso sin clasificar va para la categoría de la mayoría de sus vecinos

Ejemplo ilustrativo

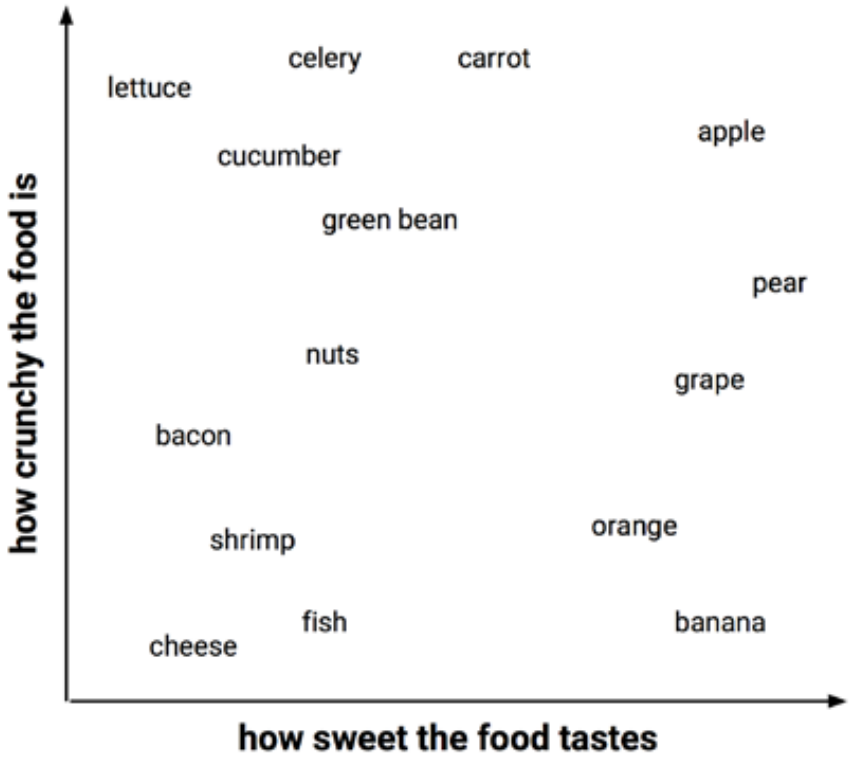
- Consideremos el siguiente ejemplo. Se venda a una persona y se le pide que pruebe un alimento misterioso
- Se le pide que clasifique el alimento. Pero antes, se ha creado una base de datos de otros alimentos
- Se identifican ciertas características de estos alimentos.
- Por simplicidad, supongamos que se registran dos características
- Qué tan crujiente es (de 1 a 10) y qué tan dulce es (de 1 a10)
- Se define a qué grupo pertenece: fruta, vegetal o proteina

Ejemplo ilustrativo

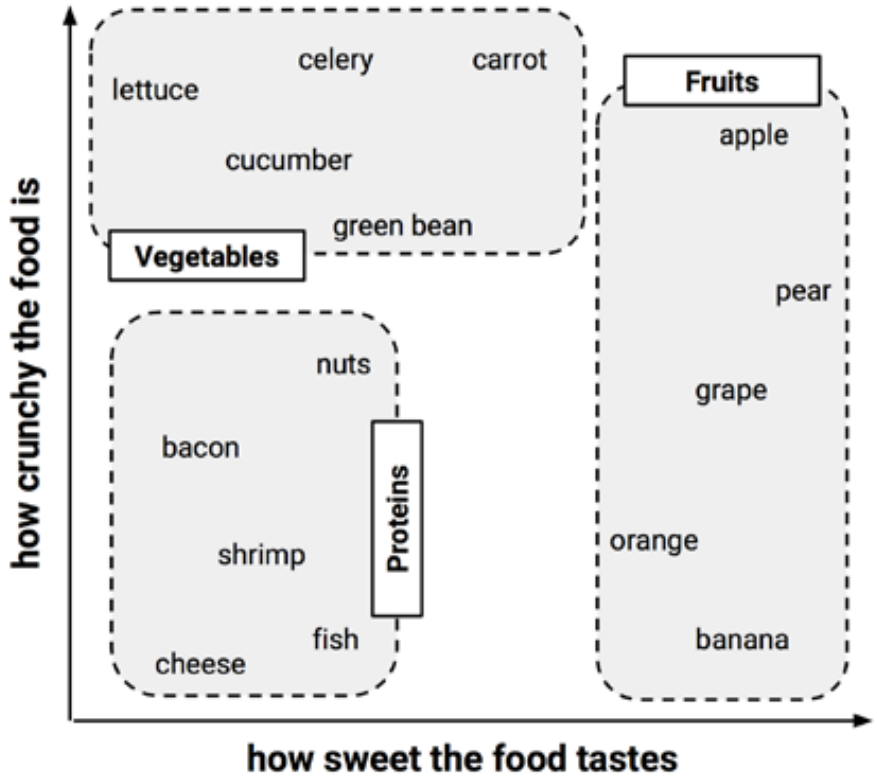
Ingredient	**Sweetness**	**Crunchiness**	**Food type**
apple	10	9	fruit
bacon	1	4	protein
banana	10	1	fruit
carrot	7	10	vegetable
celery	3	10	vegetable
cheese	1	1	protein

Ejemplo ilustrativo

¿Puede reconocerse algún patrón?

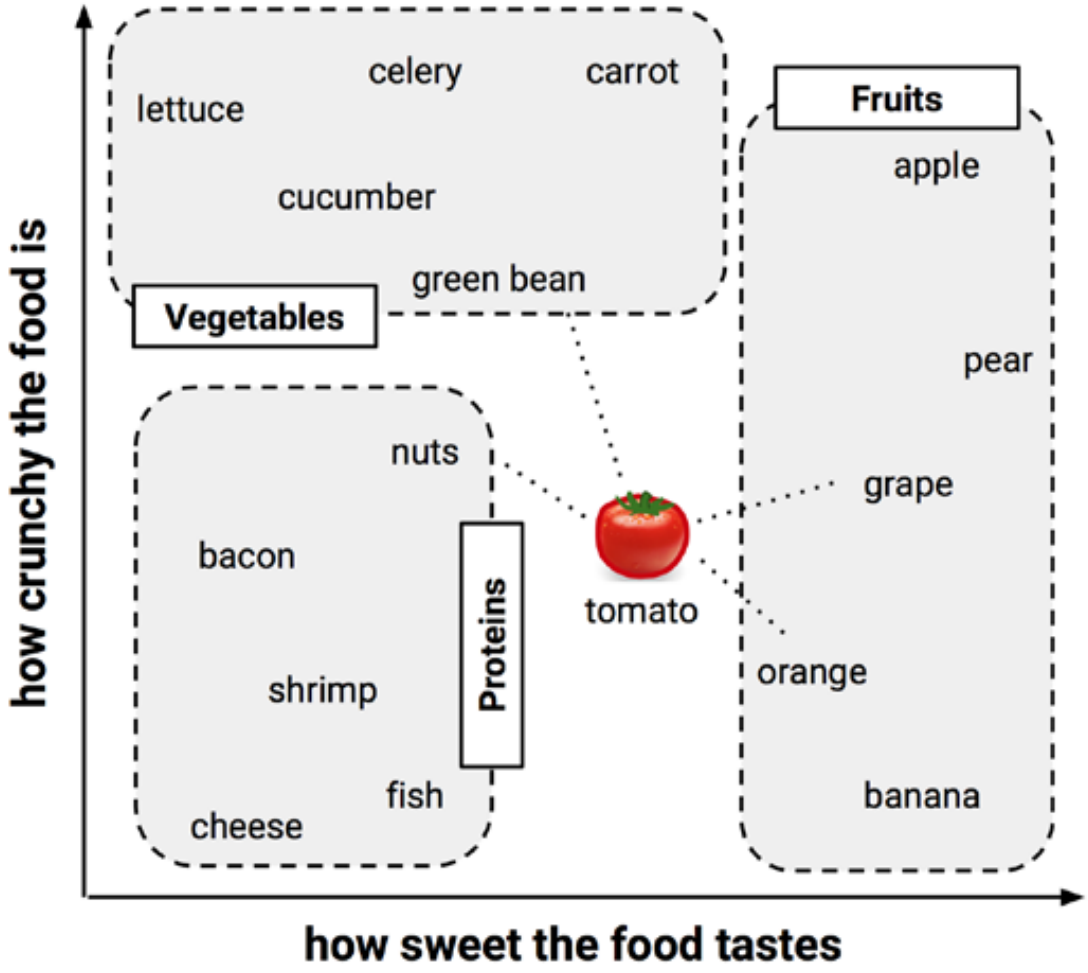


Ejemplo ilustrativo



Ejemplo ilustrativo

¿Los tomates son frutas o verduras?



Ejemplo ilustrativo

- Para determinar los vecinos más cercanos necesitamos una medida de distancia
- La más popular es la distancia euclídea entre dos vectores, que mide la longitud del segmento de recta que los conecta.

  La distancia entre dos vectores m-dimensionales, p y q es:

$$dist(p, q) = \sqrt{\sum_{i=1}^{m} (p_i - q_i)^2}$$

Ejemplo ilustrativo

- Así, en el ejemplo del tomate, calculamos la distancia a cada otro alimento

Ingredient	**Sweetness**	**Crunchiness**	**Food type**	**Distance to the tomato**
grape	8	5	fruit	*sqrt((6 - 8)^2 + (4 - 5)^2) = 2.2*
green bean	3	7	vegetable	*sqrt((6 - 3)^2 + (4 - 7)^2) = 4.2*
nuts	3	6	protein	*sqrt((6 - 3)^2 + (4 - 6)^2) = 3.6*
orange	7	3	fruit	*sqrt((6 - 7)^2 + (4 - 3)^2) = 1.4*

- El siguiente paso es definir k
- Si $k = 1$, el vecino más cercano es la naranja. Luego el tomate se clasificaría como fruta
- Si $k = 3$ los vecinos más cercanos son naranja, uva y nueces. Sería una fruta

Elección de k

- La elección de cuántos vecinos usar es crucial
- Hay un tradeoff entre sesgo y varianza
- A mayor k menor varianza generada por datos ruidosos
- Pero puede sesgar el aprendizaje al ignorar patrones pequeños pero importantes
- Si usáramos todas la observaciones todos los casos se clasificarían en la categoría mayoritaria

Elección de k

- Pero si usamos muy pocos vecinos observaciones ruidosas podrían sesgar el análisis
- Por ejemplo, si usamos solo el vecino más cercano y hay algún error de clasificación, estaríamos en problemas
- No existe una regla general de cómo elegir k.
- Una práctica común es $k = \sqrt{n}$, donde n es el número de ejemplos de entrenamiento
- En el ejemplo de los alimentos, con $n = 15$, tendríamos $k = 4$

Elección de k

- Hay que tener cuidado con estas reglas
- Lo mejor puede ser probar varios k en el conjunto de entrenamiento y quedarse con el que clasifique mejor
- Otra opción es utilizar votación ponderada
- Se le da más peso a vecinos más cercanos frente a otros lejanos

Preparación de los Datos para kNN

- Para poder calcular la distancia todas las características tienen que estar en la misma métrica
- No tendría sentido calcularla si unas características se imponen sobre otras solo por escala
- Por eso es necesario rescalar las variables cuando no están en la misma métrica
- Dos métodos:
  1. Normalización min-max
  2. Estandarización

Preparación de los Datos para kNN

1. Normalización min-max:

$$X_0 = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- Así, todos los valores están entre 0 y 1
- Se le resta el mínimo a cada valor y se divide por el rango
- Se interpreta como qué tan lejos, de 0% a 100%, está un valor dentro del rango de valores

Preparación de los Datos para kNN

2. Estandarización

$$X_0 = \frac{X - \mu}{\sigma}$$

- Se calcula el z-score de cada valor
- Mide cuántas desviaciones estándar se aleja un valor de la media
- Puede tomar valores positivos o negativos

Preparación de los Datos para kNN

- La distancia euclídea no está definida para datos nominales
- En cuyo caso creamos *dummies*. Esto es válido para dos o más categorías
- Si son n categorías, se deben crear $n - 1$ *dummies*
- Si sabemos el estado de $n - 1$ categorías, sabremos la n-ésima también
- Claramente las dummies están en la misma escala que las rescaladas min-max

kNN: Aprendizaje Perezoso

- A este algoritmo se le llama perezoso
- Esto porque no hay abstracción alguna: no se usa un modelo
- Es una técnica no-paramétrica. Esto tiene ventajas y desventajas
- No depende de un modelo predeterminado
- Pero no entendemos bien cómo se relacionan las características con el outcome a predecir