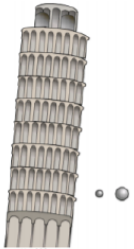
UVG UNIVERSIDAD DEL VALLE DE GUATEMALA

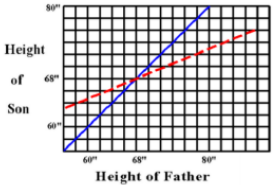
Knn, *K* VECINOS MÁS CERCANOS

Alan Reyes-Figueroa

Introducción a la Ciencia de Datos (Aula 28) 04.mayo.2022

Modelación Predictiva

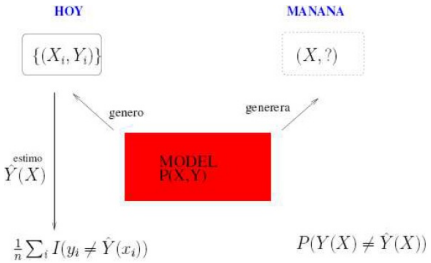




Pensamiento ha cambiado con el paso del tiempo

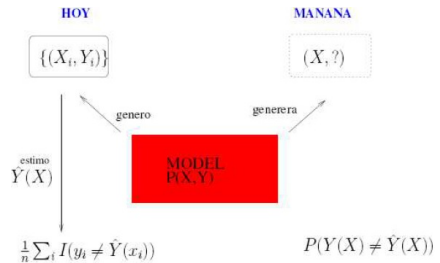
- en la antigüedad: Dioses y seres causantes del mundo
- Galileo: modelo descriptivo (separar causalidad de comportamiento)
- Hoy en día: regresión

Modelación Predictiva



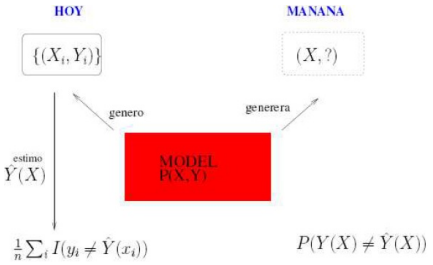
- Datos tienen dos componentes: $(\mathbf{x}_i, y_i)$, $\mathbf{x}_i$ son las variables "explicativas", y_i variable de respuesta.
- Distinguimos dos casos: $y_i \in \mathbb{R} \Rightarrow$ regresión, y_i discreto $\Rightarrow$ clasificación.
- Conectar presente con futuro: datos $(\mathbf{x}_i, y_i)$ deben ser representativos.

Modelación Predictiva



- Pragmático. Enfoque geométrico vs. enfoque probabilístico.
- Diferentes tipos de **error**: error empírico (error de entrenamiento), error de generalización (error de validación), error de prueba.
- Cada modelo tiene asociada una **complejidad**.

Modelación Predictiva

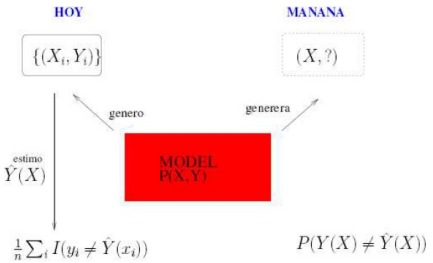


- ¿Es cierto que $\frac{1}{n}\sum_i \mathbf{1}(y_i \neq \widehat{y}_i)$ converge a $\mathbb{P}(Y(X) \neq \widehat{Y}(X))$?

  La ley (débil) de grandes números dice que

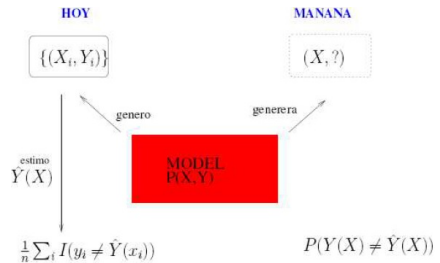
$$\lim_{n\to\infty} \frac{1}{n}\sum_i Z_i = \mathbb{E}(Z_i) = \mathbb{P}(Z_i = 1).$$

Modelación Predictiva



- Veremos que en general, no.
- Otro problema con esta función de costo empírica es que no es continua, menos diferencible.
- Si usas tu método de derivación favorito, no funciona.
  Pregunta: ¿cómo optimizar?

Modelación Predictiva



- Aprendizaje automático: imitar el aprendizaje humano.
- Historicamente: aprender o estimar con pocos datos.
- Varios tipos de aprendizaje: supervised, *unsupervised*, *semi-supervisedo*, *self-learning*, *reinforced learning*, ...

Modelación Predictiva

Tratamos de responder la pregunta

$$\frac{1}{n}\sum_i \underbrace{\mathbf{1}(y_i \neq \widehat{y}(\mathbf{x}_i))}_{Z_i} \xrightarrow[n\to\infty]{} \mathbb{P}(Y(X) \neq \widehat{Y}(X))?$$

Mencionamos que en el caso de v.a.s Bernoulli $Z_i \sim Ber(p)$, la ley de grandes números establece

$$\frac{1}{n}\sum_i Z_i \xrightarrow[n\to\infty]{} \mathbb{E}(Z) = \mathbb{P}(Z = 1).$$

¿Vale en este caso?

No.
La ley de grandes número requiere independencia de las Z_i.

Modelación Predictiva

En este caso, tenemos

$$\frac{1}{n}\sum_i \mathbf{1}(y_i \neq \widehat{y}(\mathbf{x}_i)) \xrightarrow[n\to\infty]{} \mathbb{P}(Y(X) \neq \widehat{Y}(X)),$$

donde la función $\widehat{y}$ depende de todos los datos $(\mathbf{x}_i, y_i)$ (de modo que no hay independencia de las Z_i). No aplica la ley de grandes números.

Solución *ad hoc*:
Separamos el conjunto $(\mathbf{x}_i, y_i)$ en dos:

- Conjunto de entrenamiento: lo usamos para construir la función $\widehat{y}$.
- Conjunto de validación: calculamos el error empírico $\frac{1}{n}\sum_i \mathbf{1}(y_i = \widehat{y}(\mathbf{x}_i))$. Ahora sí hay independencia, y este error empírico de validación converge al error de generalización $\mathbb{P}(Y = \widehat{Y})$.

Modelación Predictiva

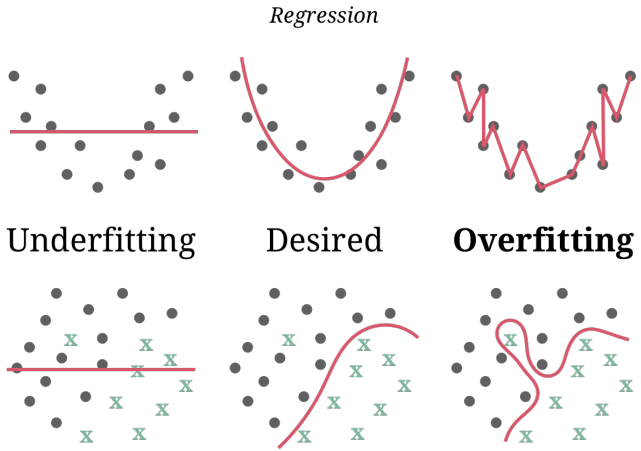
Discutimos el concepto de **complejidad** de un modelo. Este se refiere al número de parámetros involucrados en el modelo.

- en regresión: está claro, relacionado al número de variables
- en clasificación: no es tan evidente.

El concepto es importante por varias razones:

- Esto es lo que directamente va a afectar a los errores (empírico y de generalización).
- Nos va a permitir comparar diferentes modelos (en términos de simplicidad, no de exactitud).
  Veremos que existen diferentes métricas que miden la complejidad, y nos va a permitir una segunda opinión a la hora de elegir entre diferentes modelos con similar desempeño.

Modelación Predictiva



Knn, *K* vecinos más cercanos

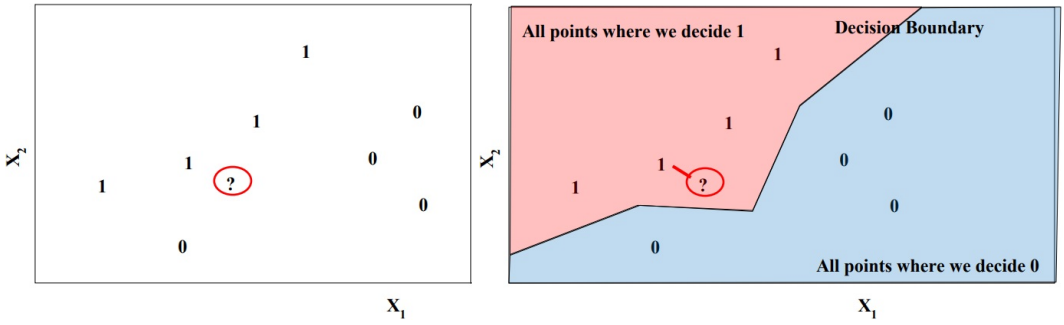
Consideramos el conjunto de datos $\{(\mathbf{x}_i, y_i)\}$, con $\mathbf{x}_i \in \mathbb{R}^d$
(en ocasiones denotamos $\mathbb{X} = (\mathbf{x}_i) \in \mathbb{R}^{n \times d}, Y = (y_i) \in \mathbb{R}^n$).

Dado $\mathbf{x} \in \mathbb{R}^d$, para decidir el valor de $\widehat{y}(\mathbf{x})$, construimos $N_k(\mathbf{x})$ el conjunto de las k observaciones más cercanas a $\mathbf{x}$.

- Para clasificación: decidimos por votación, esto es, asignamos a $\mathbf{x}$ la categoría más frequente en $\{y_i : i \in N_k(\mathbf{x})\}$.
- Para regresión: decisión por promedio, *i.e.* asignamos a $\mathbf{x}$ el promedio de $\{y_i : i \in N_k(\mathbf{x})\}$.

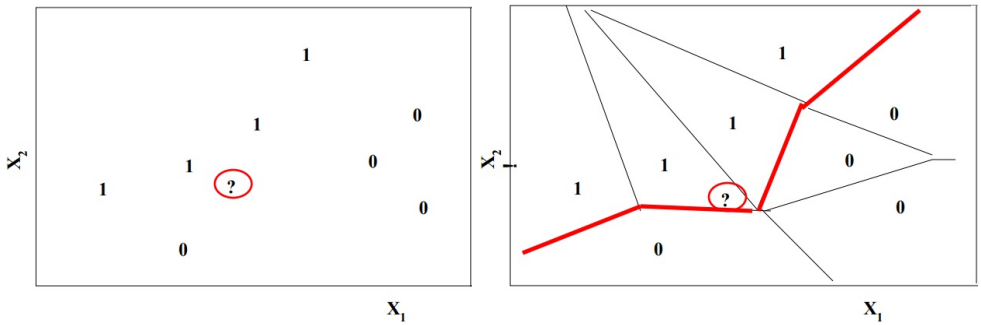
Obs! comentarios sobre cómo romper empates / métodos robustos.
El caso $k = 1$ se llama el clasificador de **vecino más cercano**.

Knn, *K* vecinos más cercanos



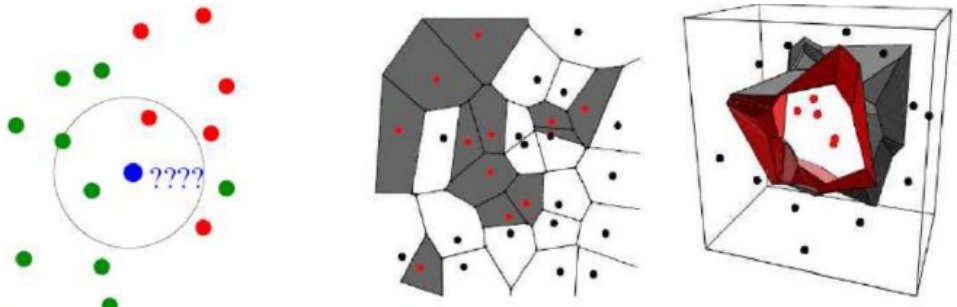
Ejemplo de *k-nn* en el caso de clasificación.

Knn, *K* vecinos más cercanos



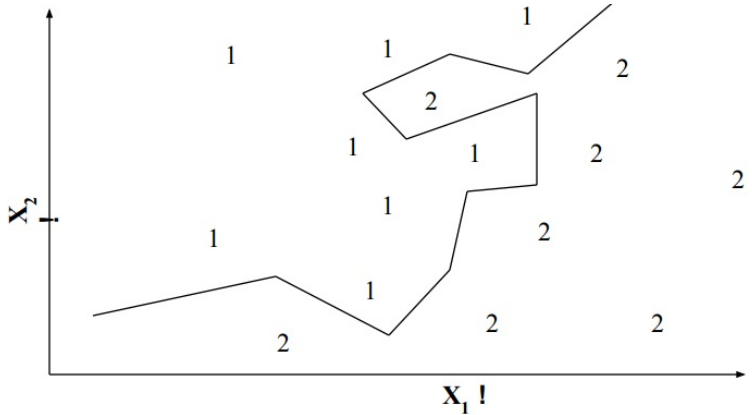
Ejemplo de *k-nn* en el caso de clasificación. Para $k = 1$, la forntera de clasificación coincide con un diagrama de Voronoi.

Knn, *K* vecinos más cercanos



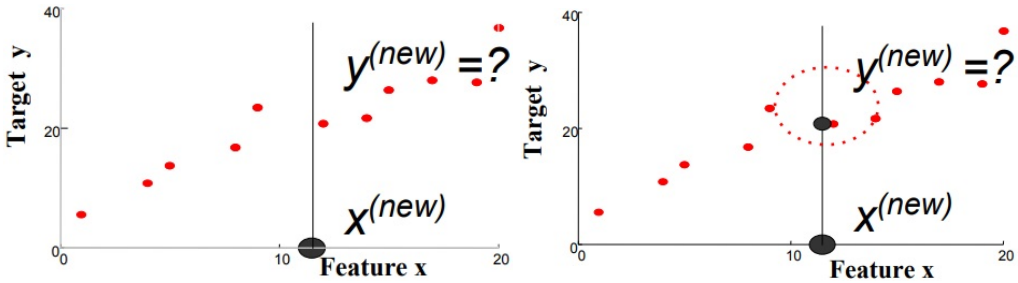
Ejemplo de *k-nn* en el caso de clasificación. Para $k = 1$, la forntera de clasificación coincide con un diagrama de Voronoi.

Knn, *K* vecinos más cercanos



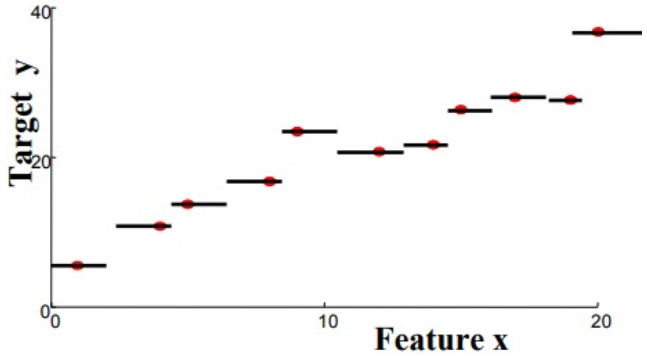
Ejemplo de *k-nn* en el caso de clasificación. En el caso general $k > 1$, la frontera sigue siendo formada por piezas poligonales (o poliedrales).

Knn, *K* vecinos más cercanos



Ejemplo de *k-nn* en el caso de regresión.

Knn, *K* vecinos más cercanos

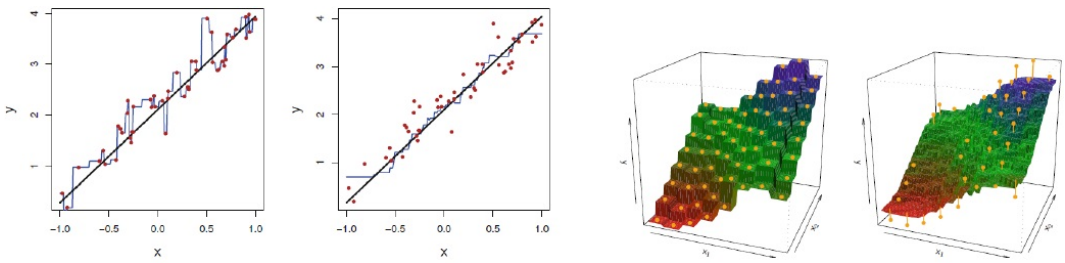


Ejemplo de *k-nn* en el caso de regresión en el caso $k = 1$. Las discontinuidades ocurren en los puntos medios entre dos observaciones consecutivas.

Knn, *K* vecinos más cercanos

Comportamiento al variar el valor de k:

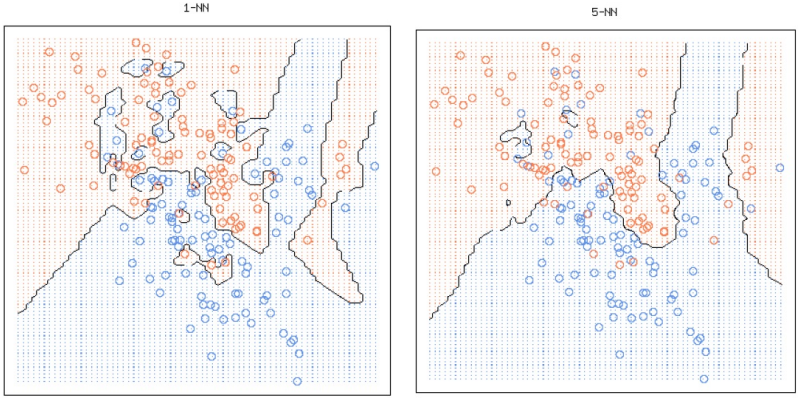
$k = 1$ $k = 7$ $k = 1$ $k = 7$



Ejemplo de *k-nn* en el caso de regresión.

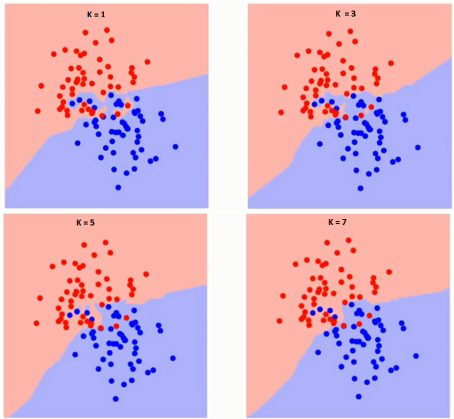
Knn, *K* vecinos más cercanos

Comportamiento al variar el valor de *k*:



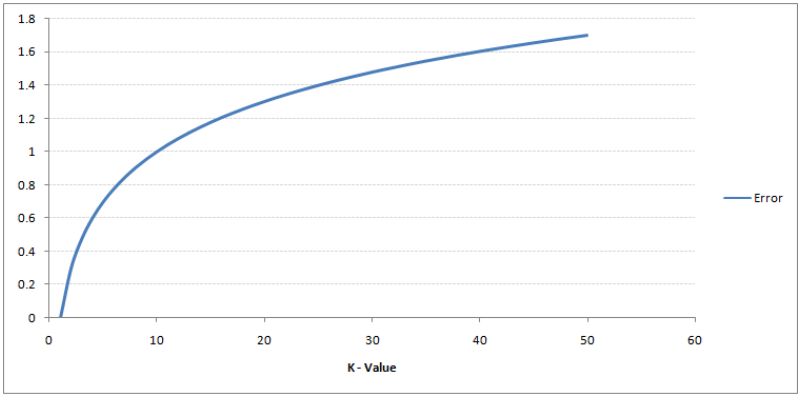
Knn, *K* vecinos más cercanos

Al aumentar *k* las fronteras de clasificación se suavizan.



Knn, *K* vecinos más cercanos

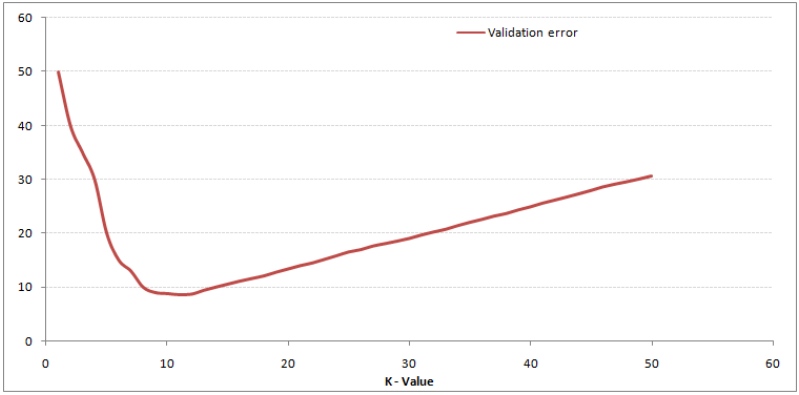
¿Cómo elegir *k*?



Error de entrenamiento en *k-nn*.

Knn, *K* vecinos más cercanos

¿Cómo elegir *k*?



Error de validación en *k-nn*.

Knn, *K* vecinos más cercanos

Knn, *K* vecinos más cercanos

Pregunta: ¿Cómo medir la complejidad en *k-nn*?