KNN: K-NEAREST NEIGHBORS

UNIVERSIDAD ICESI

KNN (K NEAREST NEIGHBORS): K VECINOS MÁS CERCANOS

- Algoritmo de aprendizaje supervisado para **clasificación** y **regresión**
- **Simple**: asignar la clase o valor agregado de las instancias conocidas que se encuentran mas cerca de la instancia a predecir
- Basado en las **instancias** de aprendizaje, no en un modelo subyacente probabilístico/estadístico
- Aprendizaje **perezoso**: en realidad el algoritmo solo se ejecuta en el momento que se requiere predecir una nueva instancia a partir de una predicción local
- Depende de la definición de una función de **distancia**, que se escogerá según la cantidad y características de las variables independientes

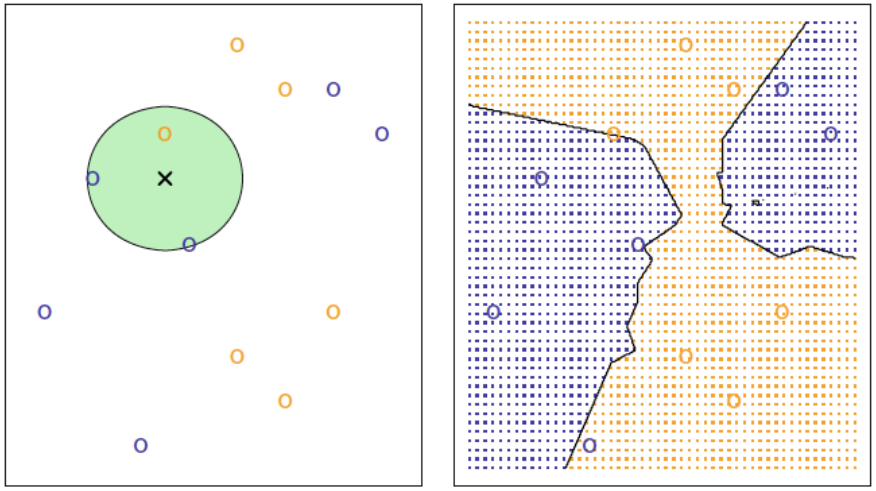
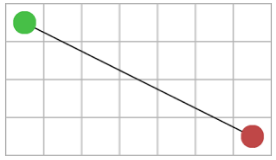


Figura 2.14 SLR

UNIVERSIDAD ICESI

KNN – DISTANCIAS

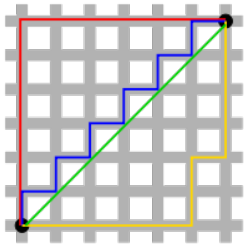
- Ejemplos de medidas de **similitud** o **distancia** utilizadas para encontrar los vecinos mas cercanos:
  - **Euclidiana**: tamaño del segmento linear que une las dos instancias comparadas.



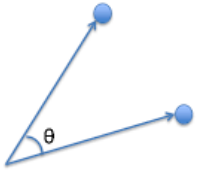
$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

  - **Manhattan**: basada en una organización en bloques rectilíneos



  - **Coseno**: coseno del ángulo entre las dos instancias comparadas → Alta dimensionalidad y **big data**

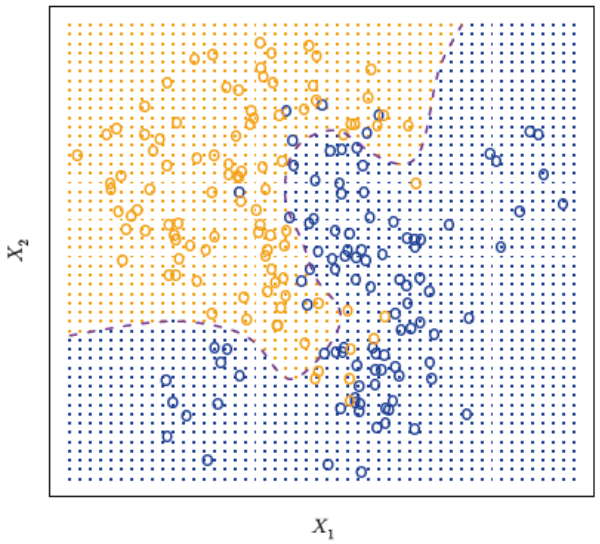


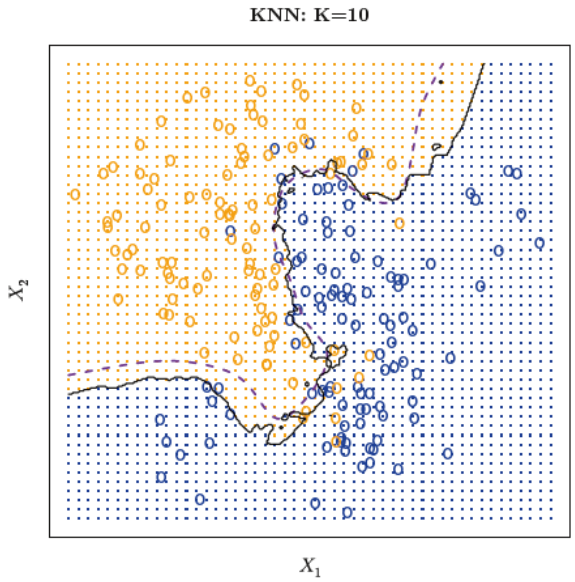
$$sim(\boldsymbol{x},\boldsymbol{y}) = \cos(\theta_{x,y}) = \frac{\boldsymbol{x}.\boldsymbol{y}}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|} = \frac{\sum_i x_i * y_i}{\sqrt{\left(\sum_i x_i * x_i\right) * \sum_i y_i * y_i}}$$

UNIVERSIDAD ICESI

KNN – K

- **Parámetro K**: número de vecinos mas cercanos a considerar para establecer la clase o valor de una nueva instancia



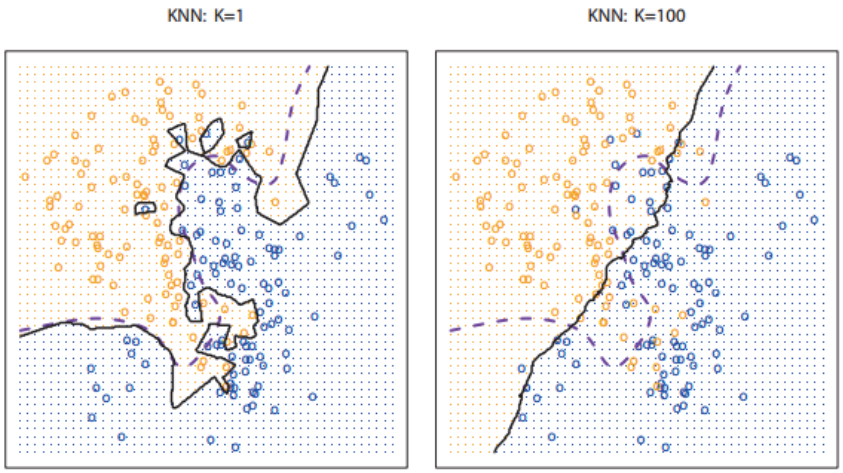


UNIVERSIDAD ICESI

James et al, ISLR, 2013

KNN – K

- **Parámetro K**
  - El resultado puede ser drásticamente diferente para diferentes valores de K
  - Un valor de K grande suavizará los límites entre clases/valores (alto sesgo, baja varianza)
  - Un valor de K pequeño resultará en límites muy flexibles (bajo sesgo, alta varianza)
  - El valor de K óptimo se encuentra empíricamente

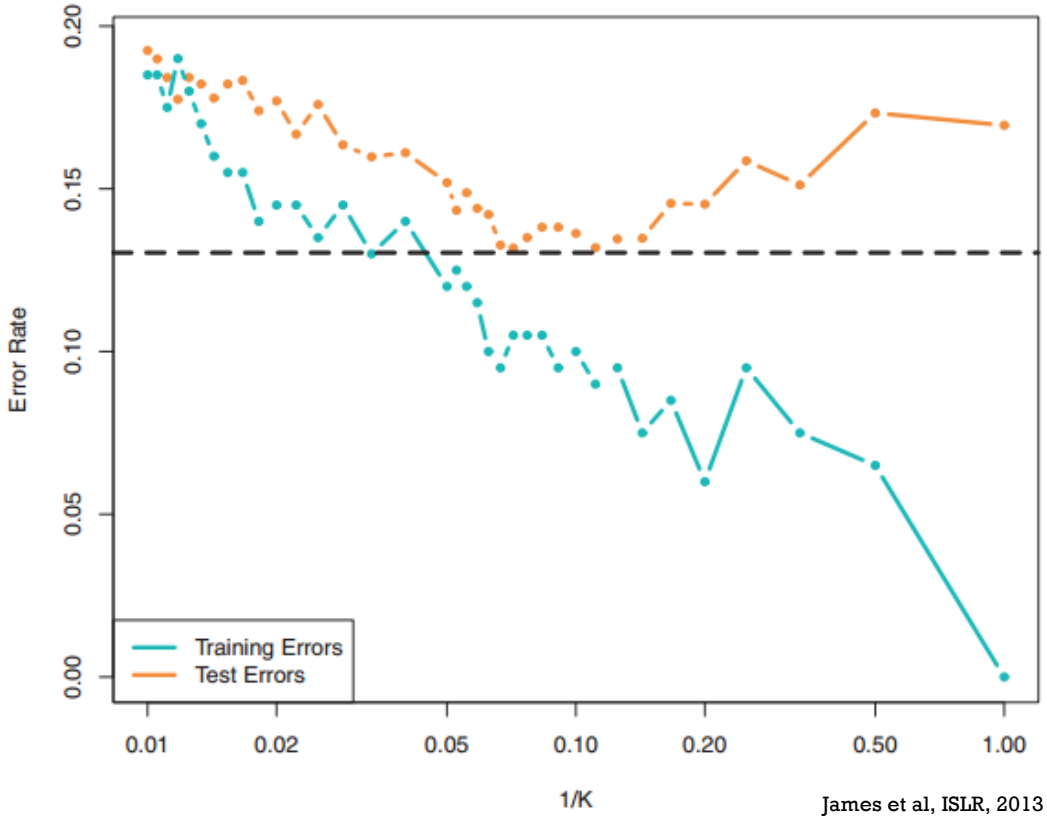


James et al, ISLR, 2013

UNIVERSIDAD ICESI

KNN – K

- K controla el **overfitting** (sobre aprendizaje) y el **underfitting** (sub aprendizaje
- Modelos mas **sencillos** (K mas grandes) previenen el overfitting, pero pueden por el contrario irse hacia el underfitting
- Modelos mas **complejos** (K mas pequeños) previenen el underfitting, pero pueden por el contrario irse hacia el overfitting
- El **K ideal** que sirva para todos los casos no existe, depende de cada dataset específico



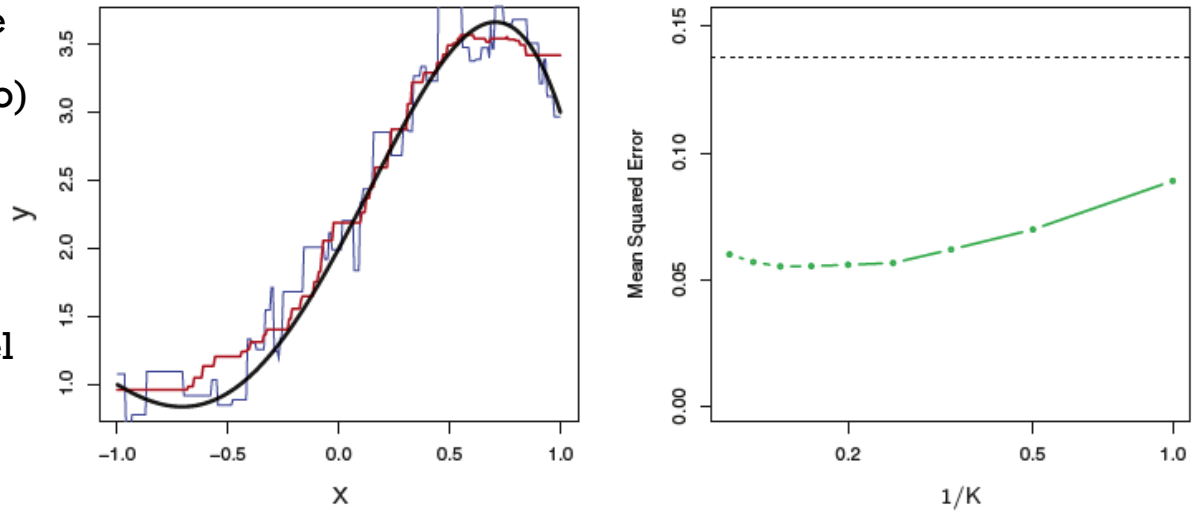
James et al, ISLR, 2013

UNIVERSIDAD ICESI

KNN – K

En el caso de la utilización de KNN para la regresión las mismas consideraciones aplican

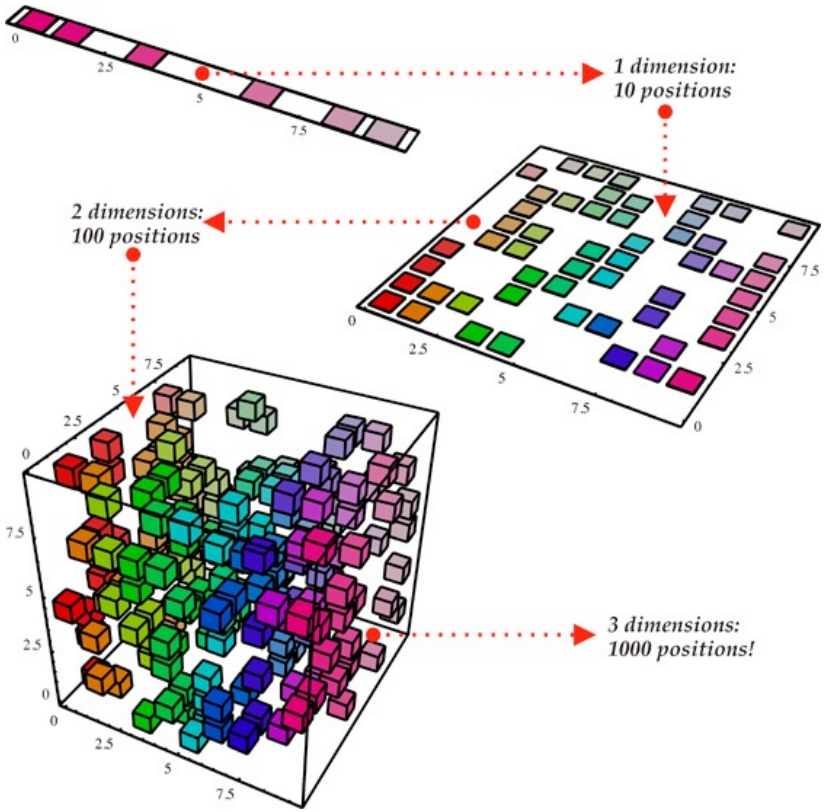
- En el panel izquierdo: se aplica KNN con un valor de K=1 (azul) y K=9 (rojo)
- En el panel derecho, se puede ver el valor de RMSE para diferentes valores de K (en verde). También se puede ver, por comparación el nivel de error de la regresión lineal simple (punteada en negro)



James et al, ISLR, 2013

UNIVERSIDAD ICESI

KNN – MALDICIÓN DE LA DIMENSIONALIDAD



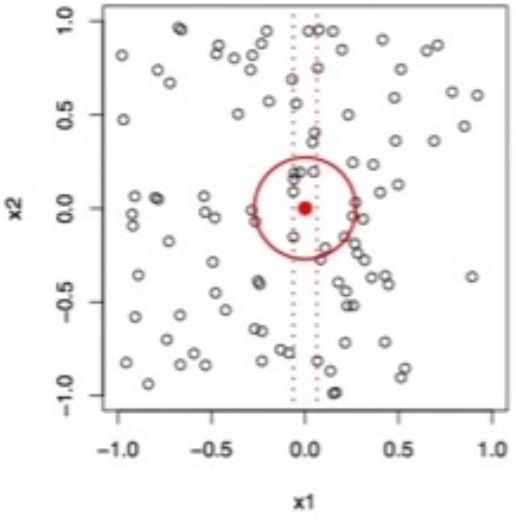
UNIVERSIDAD ICESI

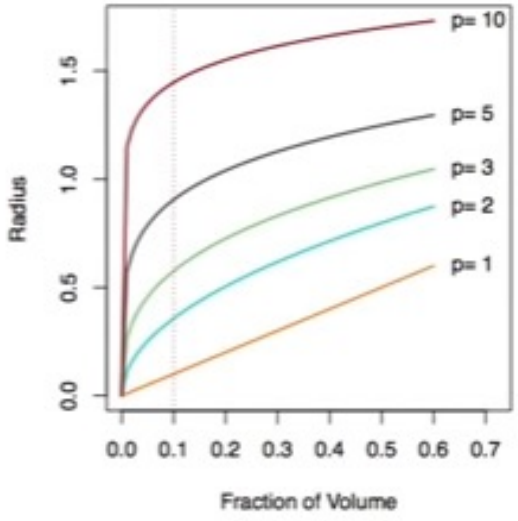
KNN – MALDICIÓN DE LA DIMENSIONALIDAD

- KNN puede llegar a ser un muy buen estimador cuando se considera un pequeño número P de variables predictivas (P <=4, con un buen número de ejemplos).
- KNN puede ser inútil cuando P es grande: todo está mucho mas lejos cuando se consideran altas dimensiones

Ejemplo: considerar el 10% de los vecinos más cercanos.

En altas dimensiones esos puntos no necesariamente son locales





UNIVERSIDAD ICESI

KNN: CARACTERÍSTICAS

- Perezoso (lazy learning), no paramétrico y no lineal
- **Método local**:
  - Puede encontrar particularidades muy específicas a ciertas regiones
  - Su uso (sobre todo en regresión) sólo permite estimaciones en los rangos de las variables del set de aprendizaje (extrapolación no tiene mucho sentido)
- Maldición de la **dimensionalidad**: no utilizar cuando el número de atributos es grande
- Al basarse en la **distancia**, es muy sensible a la **unidad de medida** de los atributos, y a atributos que no aportan poder predictivo (e.g. el color de los ojos no debería considerarse para predecir la edad de una persona)
- No sabe que hacer con los **missing values**, ni con variables **categóricas** (extensión → KnnCat)
- Complejidad temporal cuando hay **muchos registros** (extensión → CNN)