Métodos Kernel: motivación y '*kernel trick*'

Antonio Sala Piqueras

Identificación de sistemas complejos

Dept. Ing. Sistemas y Automatica (DISA)

Universitat Politècnica de València (UPV)

Presentación en vídeo en: https://personales.upv.es/asala/YT/V/kermot.html , https://personales.upv.es/asala/YT/V/ktrick.html

Presentación

Motivación:
Existen problemas técnicos con "pocas" muestras de "muchos" datos (aprender de 25 imágenes de 1Mpx, etc.).

Objetivos:
Comprender las peculiaridades de dicho tipo de problema y la regularización necesaria. Comprender que no es necesario ajustar más parámetros que datos (Kernel trick).

Contenidos:
Planteamiento del problema. Ejemplos. Generalización y regularización. Truco Kernel. Conclusiones.

¿Más regresores (q) que muestras (N)? ¡$q \geq N$!

- Complicada red bioquímica celular que relaciona x e y por un modelo con $q = 136$ parámetros, a ajustar con $N = 60$ datos.

- Procesamiento de imágenes: $N = 50$ ejemplos de letra "A" manuscrita, cada uno tiene $q = 32x32 = 1024$ pixels.

- Procesos químicos (batch): 20 mediciones/parámetros sobre composiciones y carga de materias primas. Registros cada 5 s. de 10 presiones, temperaturas, concentraciones, caudales... durante un tiempo de lote de 20 minutos (240 muestras $\times$ 10 variables). 10 medidas finales de calidad (color, humedad, densidad, composición/espectro...)... $\approx q = 2500$ medidas por lote. Se desea estimar qué desviaciones están más relacionadas con las medidas de calidad, a partir de registros históricos de $N = 180$ lotes.

Planteamiento del problema

Tenemos N datos $x_i \in \mathbb{R}^n$, $i = 1, \ldots, N$. También tenemos N variables a precedir $y_i \in \mathbb{R}^m$. Calculamos q regresores (features) $\phi(x)$, $\phi : \mathbb{R}^n \mapsto \mathbb{R}^q$.

Ejemplo: $x = (a, b) \in \mathbb{R}^2$, $\phi(x) = (a, b, 1, a^2, b^2, ab, \sin(a - \pi b), \cos(a^2 + b)) \in \mathbb{R}^8$.

Objetivo: obtener el Θ que minimiza $\sum_{i=1}^{N} L(y_i - \Theta\phi(x_i))$, siendo L una función de coste (usualmente $L(e) = e^T e$, mínimos cuadrados).

Metodología: Se generan matrices de datos $X_{q\times N} := [\phi(x_1) \ \ldots \ \phi(x_N)]$, $Y_{m\times N} := [y_1 \ \ldots \ y_N]$, y se realizan operaciones sobre ella (pinv, LP, QP,...).

Variaciones:

- Regresión **lineal** $\phi(x) = x$, vs. **no-lineal** $\phi(x) \neq x$

  [pero nos restringiremos lineal en parámetros Θ]

- Pocas "features", muchos datos ($q < N$) vs. Muchas "features", pocos datos $q \geq N$.

Ejemplo: mínimos cuadrados lineales en parámetros

Tenemos N datos $x_i \in \mathbb{R}^n$, $i = 1, \ldots, N$. También tenemos N muestras de variables a predecir $y_i \in \mathbb{R}^m$. Calculamos regresores (features) $\phi(x)$, $\phi : \mathbb{R}^n \mapsto \mathbb{R}^q$. Matrices de datos $X_{q\times N} := [\phi(x_1) \ \ldots \ \phi(x_N)]$, $Y_{m\times N} := [y_1 \ \ldots \ y_N]$, objetivo: minimizar $L(Y - \Theta X) := \|(Y - \Theta X)\|_F$.

Estos resultados son bien conocidos en álgebra lineal (ver otros materiales):

1. Si $q \leq N$ y X tiene rango (filas) completo, entonces el mejor predictor lineal ($L(\cdot) = \|\cdot\|_2^2$, minimizar $e := \|Y_{m\times N} - \Theta_{m\times q} X_{q\times N}\|_F^2$) viene dado por $\Theta = YX^\dagger = YX^T(XX^T)^{-1}$. [Pseudoinversa derecha $XX^\dagger = I$]
2. Si $q = N$, entonces $\Theta = YX^{-1}$ consigue $e = 0$. [Inversa ordinaria $XX^{-1} = X^{-1}X = I$]
3. Si $q > N$, entonces existen infinitos Θ que hacen $e = 0$, el de menos norma $\|\Theta\|_F$ es $\Theta = Y(X^TX)^{-1}X^T$. [Pseudoinversa izquierda $X^\dagger X = I$]

Problema: generalización muy mala

Incluso con menos parámetros ajustables que datos, los modelos con "demasiados" parámetros (grosso modo, $q > 0.1N$) suelen "seguir al ruido":

- Parámetros que se obtienen ante una muestra diferente de datos de entrenamiento muy alejados de los aprendidos con otras muestras.
- los márgenes de error (formalmente matriz VC de parámetros estimados) en los parámetros son muy grandes (si $q > N$, hay infinitos parámetros que ajustan "perfectamente" los datos).
- Ajuste "muy bueno" al **conjunto de entrenamiento**.
- Ajuste "penoso" en **datos de validación** separados.

Solución: Regularización

Los modelos complejos necesitan **regularización**: un balance "sensato" entre "complejidad del modelo" y "ajuste a datos".

- En vez de minimizar $L(Y - \Theta X)$, minimizamos

$$J(\Theta) = L(Y - \Theta X) + c \cdot N(\Theta),$$

siendo $N(\Theta)$ una "norma" relacionada con la complejidad (que prefiera que haya muchos elementos de Θ iguales a cero). Ej.: $\|\Theta\|_F$

***Elección de c:** Usar el valor de c que **minimice** el **error ante datos de validación**.

Eliminación de espacio nulo en $q \geq N$

Consideremos $J(\Theta) = L(Y - \Theta_{m\times q}X) + c \cdot \|\Theta\|_F$, con $q \geq N$, y $c > 0$.

- Todo Θ puede expresarse $\Theta = \Theta_X + \Theta^\perp$, con Θ_X en el espacio fila de X^T y $\Theta^\perp$ en el espacio perpendicular a él. $\Theta_X = wX^T$, $\Theta^\perp X = 0$.

*Si $q \leq N$ y $rango(X) = q$, $\Theta^\perp = 0$ [eliminación abajo innecesaria]

La descomposición en componentes ortogonales hace que

$$\|\Theta\|^2 = \|\Theta_X\|^2 + \|\Theta^\perp\|^2$$

Pero $\Theta^\perp \cdot X = 0$, por lo que
$L(Y - \Theta X) = L(Y - (\Theta_X + \Theta^\perp)X) = L(Y - \Theta_X X)$. Por tanto, **la solución óptima que minimiza $J(\Theta)$ tiene $\Theta^\perp \equiv 0$.**

- **La solución optima tiene la forma $\Theta = w_{m\times N} \cdot (X^T)_{N\times q}$.**

Combinación de filas de X^T con coefs. w.

El "truco" del Kernel

$$\min_{\Theta} J(\Theta_{m\times q}) = \min_{\Theta}\left(c\cdot traza(\Theta\Theta^T) + L(Y-\Theta_{m\times q}X)\right) \underbrace{=}_{[\Theta=wX^T]}$$

$$= \min_{w}\left(c\cdot traza(w_{m\times N}X^TXw^T) + L(Y-w_{m\times N}\cdot(X^T)_{N\times q}X_{q\times N})\right)$$

$$= \min_{w}\left(c\cdot traza(wKw^T) + L(Y-wK)\right) = \min_{w} J(w_{m\times N})$$

siendo K una matriz de productos escalares de las "features":

$$K_{N\times N} = X^TX = \begin{pmatrix} \phi^T(x_1)\phi(x_1) & \phi^T(x_1)\phi(x_2) & \cdots & \phi^T(x_1)\phi(x_N) \\ \vdots & \ddots & & \vdots \\ \phi^T(x_N)\phi(x_1) & \phi^T(x_N)\phi(x_2) & \cdots & \phi^T(x_N)\phi(x_N) \end{pmatrix}$$

A la matriz K se le denomina **Kernel matrix**.

Resultado principal

Para un número de regresores q mayor al de datos N, los "grados de libertad" en la elección de regresores ϕ son **equivalentes** a la **elección de matrices** K ("kernel", covarianza) de elementos $K_{ij} = \phi^T(x_i)\phi(x_j)$.

El problema de minimizar $J(\Theta_{m\times q})$ puede siempre reformularse a un problema **equivalente** de minimizar $J(w_{m\times N})$.

- Si $q > N$, el segundo problema tiene **menos parámetros ajustables**, igual al número de muestras N ($\times$ núm. de variables a predecir m).

Conclusiones

- En algunos problemas se deben ajustar modelos con "pocos" datos N y "muchas" variables de información (features) q por dato, $q \geq N$.
- Incluso en $q \leq N$, cuando $q > 0.1N$ suele convenir "**regularización**"; por supuesto, ello siempre es necesario con $q \geq N$.
- Con $q \geq N$, existe una versión equivalente del problema $J(Y - \Theta\phi(datos))$ con sólo N parámetros por vble. estimada (un parámetro por *muestra*). Esa versión se define en términos de matriz Kernel, $K_{ij} = \phi^T(x_i)\phi(x_j)$.

▶ **¡Truco!:** No definir $\phi(\cdot)$ "**explícitamente**", usar K entre distintas opciones en literatura, que llevan "**implícitamente**" asociado un $\phi(\cdot)$.