

Punto #2

Presente un cuadro comparativo de los siguientes regresores, en donde distinga: Modelo matemático, función de costo, estrategia de optimización, relación con los esquemas básicos de regresión dicitos en el punto 1 y escalabilidad. Linear Regressor, Lasso, ElasticNet, KernelRidge, SGDRegressor, Bayesian Ridge, Gaussian Process Regressors, Support Vector Machines Regressor, Random Forest Regressor, Gradient Boosting Regressors y XGBoost.

REGRESOR	MODELO MATEMÁTICO	FUNCIÓN DE COSTO	ESTRATEGIA DE OPTIMIZACIÓN	ESCALABILIDAD
Linear Regressor	<p>Modelo lineal: $\hat{y} = Xw$ o $\hat{y} = \phi(X)^T w$.</p> <p>Sí supone relación lineal entre entrada y salida con ruido gaussiano $\eta \sim N(0, \sigma^2)$.</p>	<p>Error cuadrático medio (MSE) $J(w) = \frac{1}{2N} \sum_{i=1}^N (y_i - X_i w)^2$</p> <p>Equivalente a máxima verosimilitud bajo ruido gaussiano.</p>	<p>Solución analítica cerrada: $w = (X^T X)^{-1} X^T t$.</p> <p>Resuelve el sistema: $X^T X w = X^T t$</p>	<p>Escala bien en problemas pequeños/medianos; en grandes versiones se prefiere una versión iterativa.</p>
Lasso	<p>Modelo lineal: $\hat{y} = Xw$ pero con regularización L_1 que promueve sparsity (coeficientes cero).</p>	<p>Equivalente a un prior laplaciano en un modelo bayesiano $J(w) = \frac{1}{2N} \sum_i (y_i - X_i w)^2 + \lambda \ w\ _1$ con $\lambda > 0$.</p>	<p>Solución numérica iterativa: <ul style="list-style-type: none"> → Descenso por coordenadas: (un paso a la vez) → Subgradiente descendiente → Least Angle Regression (construye la solución paso a paso a medida que w entra/sale). </p>	<p>Debe resolverse con métodos iterativos, lo que afecta el tiempo de procesamiento y memoria. Lento con p grande.</p>
ElasticNet	<p>Modelo lineal con mezcla de penalizaciones L_1 (Lasso) y L_2 (ridge): $\hat{y} = Xw$.</p>	$J(w) = \frac{1}{2N} \sum_i (y_i - w^T X_i - b)^2 + \lambda \left[\alpha \ w\ _1 + \frac{(1-\alpha)}{2} \ w\ _2^2 \right]$ <p>$\alpha = 1$ (Lasso), $\alpha = 0$ (Ridge) $0 < \alpha < 1$ (ElasticNet).</p>	<p>Optimización iterativa convexa</p> <p>→ Descenso por coordenadas (un paso a la vez)</p> <p>$w_j \leftarrow \frac{\sum_i z_i \lambda \alpha}{1 + \lambda(1-\alpha)}$</p>	<p>Escala bien y es ampliamente usado en datos con alta dimensionalidad ($p > n$). Puede verse afectado por λ, λ y los datos deben estar normalizados.</p>
Kernel Ridge	<p>Modelo NO lineal: $y(x) = w^T \phi(x)$ $K(x x^T) = \phi(x)^T \phi(x)$</p> <p>Extiende la regresión lineal al espacio de características de un kernel.</p>	<p>Error cuadrático con L_2: $J(\alpha) = \frac{1}{2N} \ y - K\alpha\ ^2 + \frac{\lambda}{2} \alpha^T K \alpha$, con K matriz kernel.</p>	<p>El problema es convexo y diferenciable, tiene una solución analítica cerrada: $\alpha = (K + \lambda I)^{-1} t$</p> <p>Similar formalmente a un proceso Gaussiano.</p>	<p>Adecuado solo para n pequeño/medio. Tiene mayor potencia en modelos no lineales pero su complejidad cubica limita su uso en grandes volúmenes de datos.</p>

REGRESOR	MODELO MATEMÁTICO	FUNCIÓN DE COSTO	ESTRATEGIA DE OPTIMIZACIÓN	ESCALABILIDAD.
SGDRegressor	Modelo lineal general: $\hat{y} = \mathbf{x}\mathbf{w}$ Se pone una regularización L1, L2 o ElasticNet.	Mínimiza una función de pérdida promedio, típicamente MSE o Huber: $J(\mathbf{w}) = \frac{1}{2N} \sum (t_i - (\mathbf{w}^T \mathbf{x}_i + b))^2$	Solución iterativa mediante Stochastic Gradient Descent (SGD) procesando minibatches de datos. No calcula el gradiente usando todos los datos, sino una sola muestra por iteración.	Escala linealmente con el tamaño del dataset. Excelente para datasets grandes. Requiere ajuste cuidadoso de tasa de aprendizaje.
Bayesian Ridge	Modelo lineal bayesiano: $\hat{y} = \mathbf{x}\mathbf{w} + \boldsymbol{\eta}$, $\mathbf{w} \sim N(0, \lambda^{-1}\mathbf{I})$, $\boldsymbol{\eta} \sim N(0, \alpha^{-1}\mathbf{I})$. No asume un valor para \mathbf{w} y $\boldsymbol{\eta}$ sino distribuciones para cada uno.	Maximiza la evidencia del modelo equivalente a MAP con prioris gaussianos: $\log p(\mathbf{w} \mathbf{t}) = -\frac{B}{2} \ \mathbf{t} - \mathbf{x}\mathbf{w}\ ^2 - \frac{\alpha}{2} \ \mathbf{w}\ _2^2$ \mathbf{x} y B no son fijos, son variables aleatorias.	Solución analítica usando teoría bayesiana, dado que las distribuciones son gaussianas.	Combinación probabilidad con costo razonable, limitado en datasets grandes. Aporta estimaciones de incertidumbre sobre los parámetros y las predicciones.
Gaussian Process Regressor	Modelo no paramétrico: $f(\mathbf{x}) \sim NP(0, K(\mathbf{x}, \mathbf{x}'))$, $\hat{y} = f(\mathbf{x}) + \boldsymbol{\eta}$. Es una extensión de la regresión bayesiana lineal. La función está definida por su kernel	Se ajustan los hiperparámetros del kernel y del ruido: $\log P(t \mathbf{x}) = -1/2 \mathbf{t}^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{t} - 1/2 \log \mathbf{K} + \sigma_n^2 \mathbf{I} - N/2 \log (2\pi)$	2 etapas: a.) Inferencia: Distribución posterior: $p(t_n \mathbf{x}, t, \mathbf{x}') = N(f_n, var(t))$ $f_n = k_{\mathbf{x}'}^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{t}$. b.) Optimización de hiperparámetros mediante gradiente descendente.	Es el más completo de la familia bayesiana, pero solo es bueno para conjuntos medianos. Tiene incertidumbre real.
Support Vector Machines Regressor	Modelo kernelizado: $\hat{y}(\mathbf{x}) = \sum_i (a_i - a_i^*) K(\mathbf{x}, \mathbf{x}_i)$	Minimiza $\frac{1}{2} \ \mathbf{w}\ _2^2 + C \sum_i (\xi_i + \xi_i^*)$ $\xi_i \rightarrow$ errores que exceden el margen.	Se usan multiplicadores de Lagrange: $\max_{\mathbf{d}} -\frac{1}{2} \mathbf{d}^T (\mathbf{d} - \mathbf{d}^*)^T + \sum_i (\alpha_i + \alpha_i^*)$ $+ \sum_i \alpha_i (\mathbf{d}_i - \mathbf{d}_i^*)$. Se obtiene la solución usando método de optimización cuadrática.	La versión lineal se usa para muchos datos, pero kernel SVR solo para datasets medianos.
Random Forest Regressor	Modelo de ensamble formado por B árboles de decisión independientes: $f(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B f_b(\mathbf{x})$ (Cada árbol predice un valor y se toma la media).	Cada árbol se entrena minimizando el MSE = $\frac{1}{N} \sum_{i=1}^N (t_i - y_i)^2$ Se elige la N característica y el punto de corte: $\text{split}(y, s) = \arg \min [\text{MSE}_{\text{izq}} + \text{MSE}_{\text{der}}]$	Tiene 2 modo de aleatoriedad controlada: 1. Bootstrap Sampling 2. Feature Bagging 3. Crecimiento de Árboles 4. Agregación (Media Final).	Escalable, paralelizable y resistente al ruido, puede estimar incertidumbre por variación entre árboles.

REGRESOR	MODELO MATEMÁTICO	FUNCIÓN DE COSTO	ESTRATEGIA DE OPTIMIZACIÓN	ESCALABILIDAD
Gradient Boosting Regressor	<p>El modelo es una suma ponderada de árboles débiles:</p> $f(x) = \sum_{m=1}^M h(x)$ <p>$M \rightarrow$ número de iteraciones</p> <p>$h(x) \rightarrow$ Árbol de decisión entero</p> <p>$\lambda \in [0, 1]$: learning rate</p> <p>$f(x)$: predicción final.</p>	<p>Trabaja con cualquier función de pérdida diferenciable</p> $L(y, f(x))$. En regresión clásica: $L(y, f(x)) = \frac{1}{2} (y - f(x))^2$. <p>Minimiza la $\frac{1}{2}$ pérdida total (todo el dato).</p>	<p>Aplica el descenso del gradiente, pero en el espacio de las funciones, no en el de los parámetros.</p> <ol style="list-style-type: none"> 1. Función que minimizar. 2. Gradientes Negativos 3. Ajuste del árbol al residuo. 4. Peso optimo del árbol 5. Actualizar modelo 6. Repetir M iteraciones. 	<p>El costo depende del número de árboles M y de muestras N.</p> <p>No es bueno cuando se tiene $N > M$, pero es robusto y tiene mayor sensibilidad.</p>
XG Boost	<p>Es un modelo aditivo como el GBR, pero añade regularización explícita y una aproximación de 2do orden al gradiente</p> $f(x) = \sum f_m(x), f_m \in F$ <p>$F \rightarrow$ conjunto de árboles</p> <p>$M \rightarrow$ Número de iteraciones,</p>	<p>En cada iteración, minimiza</p> $L = \sum_{i=1}^n (y_i - \hat{y}_i) + \Omega(f_m)$ <p>$L(y_i, \hat{y}_i)$ es la función de pérdida.</p> <p>$\Omega \rightarrow$ es la penalización sobre la complejidad del árbol.</p>	<p>Utiliza un desarrollo de 2do orden de la pérdida:</p> $L(y_i, \hat{y}_i) + f_m(x_i) \approx L(y_i, \hat{y}_i) + g + f_m(x_i) + \frac{1}{2} h \cdot f_m(x_i)^2$ <p>$g \rightarrow$ gradiente</p> <p>$h \rightarrow$ Hessiana.</p>	<p>Es eficiente y robusto, logra suavidad local y escalabilidad masiva.</p>

A continuación, el cuadro comparativo de relación entre los regresores y los ejemplos bárticos discutidos en el 1er punto.

REGRESOR	MÍNIMOS CUADRADOS	MÁXIMA VEROSIMILITUD	MÁXIMO POSTERIOR	BAYESIANO LINEAL	REGRESIÓN RIGIDA KERNEL	PROCESOS GAUSSIANOS
Linear Regressor	Es la implementación directa del modelo de mínimos cuadrados.	Equivalente a ML si se asume ruido Gaussiano con varianza constante.	Introduce creencia previa. Se convierte en "ridge regressor" al agregar regularización.	Considera toda la distribución sobre los pesos:	Espacio de características no lineal.	Modelar función (como distribución gaussiana $f(x) \sim \mathcal{N}(0, K(x, x))$)
Lasso	Variante del lineal regresor con penalización L1 (no diferenciable).	la función de costo puede verse como una verosimilitud max regularizada:	Introduce un prior laplaciano sobre w	$p(w t) = \mathcal{N}(w 0, S_n)$ $p(w) = \mathcal{N}(0, \alpha^{-1} I)$	$\min_w \frac{1}{2} \ t - \Phi w\ ^2 + \frac{\lambda}{2} \ w\ ^2$. Para lineal regresor: kernel = $x x^T$. Kernel lasso:	$\min_w \frac{1}{2} \ t - \Phi w\ ^2 + \lambda \ w\ _1$ El prior ya no es gaussiano. Si se define un ruido con ruido laplaciano, se puede definir como lasso.

REGRESOR	MÍNIMOS CUADRADOS	MÁXIMA VEROSIMILITUD	MÁXIMA A POSTERIORI	BAYESIANO LINEAL	REGRESIÓN RIGIDA KERNEL	PROCEJO GAUSSIANOS
ElasticNet	El modelo matemático parte de mínimos cuadrados añadiendo L_1 y L_2 .	El segundo término actual compuesta penalización bayesiana hibrida (Laplaciana+gaussiana)	$-\log p(t w) =$ Proviene del ruido gaussiano $\log p(w) =$ Prior mixto ($L_1 + L_2$).	$p(w) = e^{-\alpha \ w\ _1} \ w\ _2^2$ $+ (1-\alpha) \ w\ _2^2$	Kernel Elastic Net: $J(w) = \frac{1}{2} \ t - \phi w\ _2^2$ $+ \gamma [\alpha \ w\ _1 + 2 \ w\ _2^2]$	Se puede considerar como una mezcla de gaussiana y regularizada.
Kernel Ridge	Versión no lineal y regularizada. Introduce transformaciones no lineales de las variables.	Espacio de características $\phi(X_n)$. $t_i = w^\top \phi(x_i) + \epsilon \sim N(0, \alpha^{-1} I)$. El log Zeta en L_2 .	$p(t w) = N(\phi w, \beta^{-1} I) \rightarrow p(w) = N(w M_h, S_h)$ Posterior es: $y w) = \beta/2 \ t - \phi w\ _2^2 + \alpha/2 \ w\ _2^2$	$S_h^{-1} = \alpha I + \beta X^\top X$ $M_h = P_{f,h} X^\top I$ X se reemplaza por la matriz de características $\bar{x} = \alpha/\beta$: Función costo.	Puede operar en espacios no lineales mediante el kernel.	Tiene la misma fórmula de $p(t x) = N(0, K^{-1} G^2 I)$ pero si es tratado como un valor determinista.
SGD Regressor	Minimiza la mínima función de costo, pero iterativamente usando gradientes por muestra: $w \leftarrow w - \eta \nabla J(w)$	Implementa aproximación estocástica actuando parámetros de forma progresiva	Incluye: Prior Gaussiano Prior Laplaciano Prior Mixto Se adapta según el regresor.	Obtiene solo el valor más probable mediante actualizaciones sucesivas.	Trabaja en el espacio original de las características y no usa kernels.	Se puede considerar como una versión lineal de GP.
Bayesian Ridge	Incluye regularización empírica probabilística: $J(w) = \frac{\beta}{2} \ t - x_m\ ^2 + \alpha p(w B)$	La inferencia sobre $p(w)$ es probabilística: $p(w t, \alpha, \beta)$	Adapta una distribución gaussiana sobre $p(w)$.	Versión paramétrizada del modelo donde α y β se estiman de los datos directamente	$J(w) = \beta/2 \ t - x w\ _2^2 + \alpha/2 \ w\ _2^2$	Se considera un GP con un espacio de características que obtiene distribuciones de incertidumbre sobre los pesos.
Gaussian Process Regressor.	Define una distribución sobre funciones. Entrega una familia finita de curvas con media y varianza. $f(x) \sim N(0, K(x, x'))$ sin prior ni incertidumbre.	No hay parámetros fijos, se maximiza la verosimilitud marginal sobre las funciones.	Se integra sobre el espacio de funciones completo.	Define una matriz de covarianza $K = x x^\top$. Cuantifica la incertidumbre.	Es el bayesiano ridge con infinitas bases implícitas.	La media es igual al resultado del Kernel Ridge; pero sin incertidumbre.

①

REGRESOR	MÍNIMO (CUADRADO)	MÁXIMA VEROSIMILITUD	MÁXIMO A POSTERIORI	BAYESIANO	REGRESIÓN RIGIDA	PROCESES GAUSSIANO
				LÍNEAL	KERNEL	
Support Vector Machine Regressor	Minimiza la norma de los pesos y penalidad solo los errores mayores a ϵ .	Tiene máxima verosimilitud con ruido laplaciano truncado. Tiene zona de tolerancia.	Usa una función de pérdida lineal por tramas.	Busca modelar el ruido con una toleran- cia tipo.	Busca minimizar la pérdida Q . $f(x) = \sum (a_i - a_i^*) K(x, x_i) + b$.	Fs de naturaleza determinista y no se considera gaussiana, solo trabaja con la media.
Random Forest Regressor	Ajusta múltiples funciones locales, cada una valida en una región del espacio. Puede aproximar regiones no lineales.	No reduce gli errores Maxima Verosimilitud reduce la varianza asume mediatruido gaussiano i.i.d.	No es un estimador MAP clásico. Es un ensamble no paramétrico de MSE por nodo.	La incertidumbre que se reporta suele ser empírica no una varianza bayesiana bien calibrada.	Se puede ver como una regresión kernel pues dos puntos pueden tener similaridad si caen en la misma hoja y tener un mismo núcleo.	Comparte la idea de promediar puntos similares, pero no hay pesos ni posterior probabilístico
Gradient Boosting Regressor	Minimiza $\sum L(y, f(x_i))$ por gradiente funcional y tiene solución iterativa. Tiene regularización implícita (con el número de árboles).	Minimiza la perdida cuadrática, que es equivalente a maximizar la verosimilitud. Realiza descenso del gradiente sobre el log-likelihood.	No asume ningún prior ni probabilidad, pero si regulariza implíci- tamente. El learning rate es similar como un prior suave.	Ambos son promedios ponderados, pero el GBR es iterativo sobre un modelo flexible.	Tiene linealidad a través de los árboles y suma los árboles que (origen errore), como los kernels.	Ambos construyen modelos de funciones no lineales, pero el GBR lo hace determini- sticamente. Si es Kernel tiene regiones locales, su media se importa similar al ajuste por boosting.
XG Boost	Si se usa $L(y, \hat{y}_i) = \frac{1}{2} (y - \hat{y})^2$ $y^T \pi = 0$, es un GBR, pero con regularización activa como la Ridge NO lineal.	Minimiza perdidas equivalentes a log-likelihood negativos, pero la regularización explicita π , convierte el entrenamiento en un MAP con un modelo no lineal.	Es una implementación determinista del MAP, donde el prior está (definido en los hiperpárametros) (π, y, α) .	Cada hoja tiene una creencia previa de que los valores deben ser pequeños.	Es una red discretizada, jerárquica y regularizada del Kernel, donde los árboles reemplazan los kernels.	No usa kernels explícitos, pero cada árbol tiene regiones locales donde se agrupan puntos similares, no tiene varianzas, ni posterior gaussiano.