Trabajo Práctico N°4 - Ciencia de Datos

Clasificando pobres en la EPH: Métodos de regularización y CART

Grupo 10

Francisca Cetra, Mariano Ripoll y Justina Rivero Ayerza

Profesora Magistral

María Noelia Romero

Profesor Tutorial

Tomas Enrique Buscaglia

Semestre y año de presentación

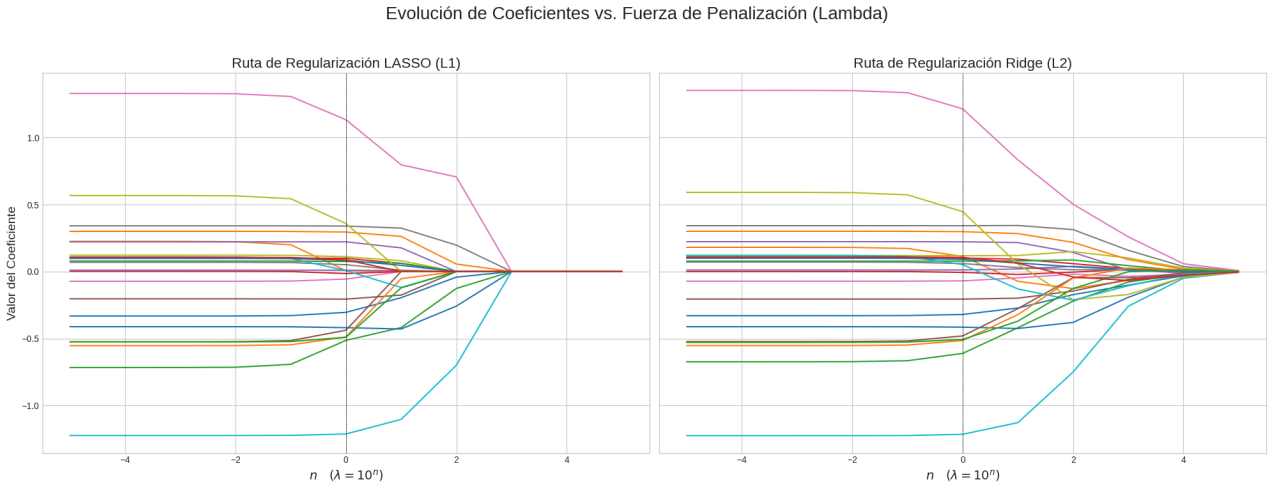
2do Semestre 2025

Repositorio de Github: https://github.com/JustinaRiveroA/Ciencia_Datos_TP4_Grupo10

A. Modelo de Regresión Logística: Ridge y LASSO

1. Visualización

Figura 1. Evolución de rutas de coeficientes vs. fuerza de penalización (LASSO vs. Ridge)



El eje horizontal muestra los valores del exponente n pertinentes a la función del parámetro de penalidad ($\lambda = 10^n$), acotados en enteros de -5 a 5. El hiper parámetro C es el inverso de la penalización ($C = 1/\lambda$), por lo que moverse hacia valores mayores de λ equivale a mayor regularización.

En el panel de LASSO (L1) se ve el patrón típico: cuando la penalización aumenta, varios coeficientes se acercan a cero y algunos quedan exactamente en cero, lo que refleja selección de variables. Para valores de n mayores a cero, a medida que la penalización aumenta (cruzando la línea vertical punteada de n=0), los cambios en el modelo aumentan también. Las líneas que comienzan a tender a cero muestran que los coeficientes no solo se reducen, sino que (en aquellos que se cruzan con el eje horizontal) son forzados a convertirse en cero. Puede verse que para n=2 donde lambda es 100, varias variables ya fueron descartadas del modelo. Además, para valores de n mayores a 3, todo coeficiente converge a cero. Vemos así un evidente exceso de penalización a partir de estos valores.

En Ridge la dinámica de contracción de los coeficientes es bastante distinta: las curvas son más suaves y menos bruscas y más prolongadas para valores de n. Tienden a valores próximos a cero para valores crecientes de n, pero no se anulan tan repentinamente

como con LASSO. De este modo, incluso en penalizaciones altas con n mayor a 3, aún hay coeficientes relevantes seleccionados ya que distribuye la penalización entre todas las variables y mantiene una estructura más completa de los predictores.

2. Penalidad Óptima por Cross Validation

Necesitamos quedarnos con el λ que mejor generaliza fuera de muestra. Para eso usamos validación cruzada de 5 folds sobre el set de entrenamiento. En la práctica, para cada λ de la grilla $\{10^{-5}, \dots 10^{5}\}$ entrenamos el modelo cinco veces (cada vez dejando un fold distinto como validación) y obtenemos cinco errores de validación. Esos cinco valores se resumen con boxplots, lo que permite ver no solo el nivel del error típico (mediana) sino también su variabilidad entre folds.

El procedimiento seleccionó el mismo valor óptimo para ambos modelos: para LASSO (L1) y para Ridge (L2) el óptimo fue C=0.1, lo que equivale a $\lambda = 10$. Esto sugiere que el problema principal no era que un modelo necesitara selección agresiva y el otro no, sino que en general conviene imponer una penalización intermedia/alta para estabilizar el Logit en este set de predictores de EPH. En un dataset como el de la EPH, donde muchas variables capturan dimensiones parecidas (por ejemplo, distintas medidas de inserción laboral, variables demográficas relacionadas y transformaciones), la regularización ayuda a evitar que el modelo se apoye demasiado en variaciones específicas del entrenamiento.

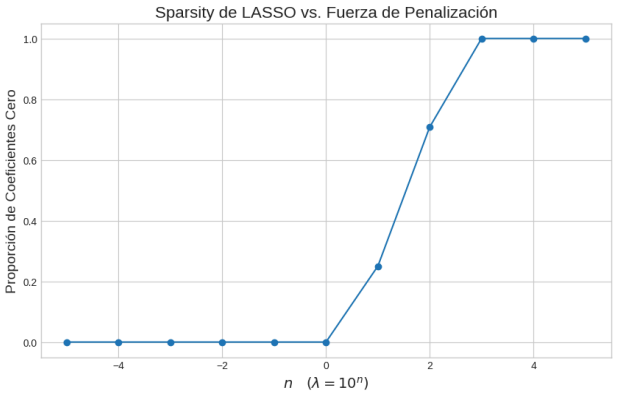
Figura 2. Boxplots Error de validación cruzada para LASSO (L1) y Ridge (L2).

Al interpretar la figura, se observa que para penalizaciones muy bajas (valores pequeños de λ) el error se mantiene en niveles razonables, pero no alcanza el mínimo. La

mejora más clara aparece alrededor del valor seleccionado por CV: con λ=10 la mediana del error es la más baja y el desempeño promedio en validación se ubica cerca de 0.789 de accuracy para LASSO y 0.787 para Ridge. Además, la dispersión de los errores entre folds se mantiene acotada, lo cual es importante porque indica un resultado relativamente estable: no depende de una sola partición favorable del entrenamiento.

En contraste, cuando λ crece demasiado (penalizaciones extremas), el desempeño cae. En particular, para valores muy altos como λ=10000 el accuracy baja aproximadamente a 0.58, lo que refleja un escenario de subajuste: la penalización es tan fuerte que el modelo pierde capacidad de usar la información de los predictores y termina clasificando de manera mucho menos precisa. Esta caída es coherente con lo que se espera teóricamente: regularizar demasiado aplana el modelo y lo vuelve incapaz de captar diferencias relevantes entre pobres y no pobres.

Figura 3. Sparsity de LASSO vs. Fuerza de Penalización



Para complementar un poco lo anterior, analizamos la *Sparsity*: la proporción de coeficientes que quedan exactamente en cero según λ. Este gráfico es útil porque muestra, de manera directa, el trade-off entre *simplificación del modelo* y *capacidad predictiva*.

Cuando λ es baja, LASSO casi no elimina variables (Sparsity ≈ 0). A medida que λ empieza a subir, aumenta la proporción de coeficientes en cero: el modelo se vuelve más fácil de explicar. Sin embargo, si la Sparsity se hace muy alta, se corre el riesgo de perder variables relevantes y empeorar el desempeño. Por eso, este gráfico no reemplaza a la validación cruzada. Muestra "qué precio en complejidad" estamos pagando o ahorrando con el λ elegido.

3. Estimación Final e Interpretación de Coeficientes

Tabla 1. Coeficientes sin penalidad vs. LASSO vs. Ridge

Variable	Logit (Sin Pen)	LASSO (λ=10)	Ridge (λ=10)
Educación	-0.329788	-0.195316	-0.272536
Edad	0.180624	0.000000	-0.072832
Edad2	-0.673921	-0.418724	-0.419856
Ocupados Share	-1.225896	-1.104181	-1.128498
Horas Trabajadas (PP3E_TOT)	-0.072203	0.000000	-0.047500
Categoría Ocupacional	-0.523030	0.000000	-0.277085
Cobertura médica: Obra Social	0.591112	0.000000	0.055479
Cobertura médica: No Paga	1.353104	0.797142	0.833825

Lo primero que se observa en la tabla es el patrón general de shrinkage: los coeficientes de los modelos regularizados tienden a ser menores en magnitud absoluta que los del Logit sin penalidad. Esto es esperable porque la regularización "castiga" coeficientes grandes para reducir varianza y evitar sobreajuste. Por ejemplo, el coeficiente de Educación mantiene el signo negativo (más educación se asocia a menor probabilidad de pobreza) pero con un efecto atenuado cuando priorizamos robustez fuera de muestra).

También vemos cómo aparece la no linealidad de la edad. En el Logit sin penalidad, Edad (CH06) entra con signo positivo (0.1806), pero en LASSO queda exactamente en cero, y en Ridge incluso aparece con signo negativo (−0.0728). En cambio, el término Edad2 se sostiene en ambos regularizados con coeficientes negativos similares. Esto significa que Edad está absorbida por Edad2 y/o por otras variables que también capturan ciclo de vida y composición del hogar.

Al penalizar el modelo "decide" que el componente realmente informativo es el cuadrático (relación curvilínea) y que la edad lineal agrega una redundancia que no ayuda a generalizar. En particular, LASSO directamente se queda con Edad2 y elimina el término lineal.

En variables laborales, el resultado más consistente es que Ocupados Share es extremadamente robusta. Su coeficiente es grande y estable en los tres modelos. Esto sugiere que la proporción de ocupados en el hogar captura una dimensión estructural central (capacidad del hogar para generar ingresos) que sobrevive incluso cuando penalizamos fuerte. En cambio, otras variables laborales se comportan como redundantes cuando se introduce LASSO. La variable de Horas Trabajadas aparece negativa en Logit y en Ridge, pero LASSO la lleva a cero exacto. Una interpretación es la colinealidad: horas trabajadas y categoría ocupacional suelen estar muy relacionadas con estar ocupado, y con el proxy agregado Ocupados Share. En este sentido, LASSO elige un resumen fuerte del mercado laboral del hogar y descarta desagregaciones que no agregan poder predictivo fuera de muestra.

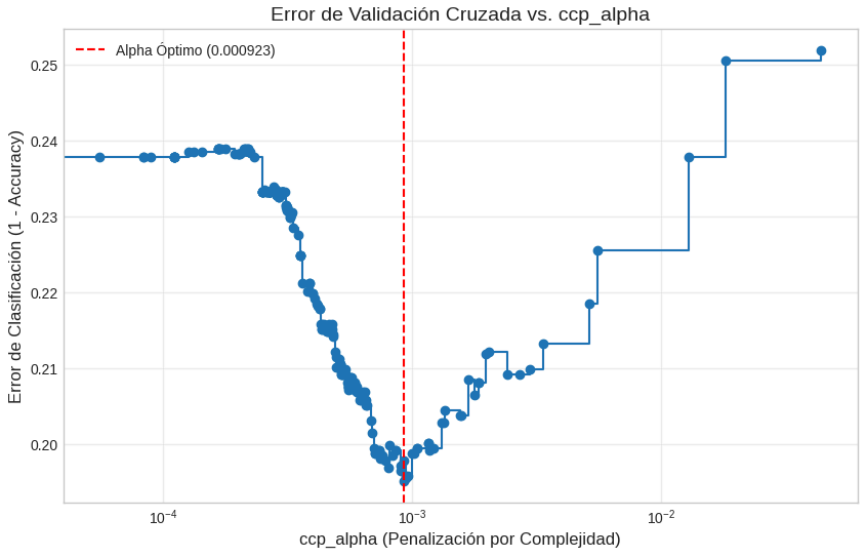
En variables de salud aparece una distinción bastante clara. Obra social es positiva en el Logit sin penalidad (0.5911) y casi nula en Ridge, pero LASSO la elimina. En cambio, Cobertura médica: No paga se mantiene como una señal fuerte en los tres modelos. Parecería que "No paga" queda como indicador robusto de vulnerabilidad, mientras que "Obra social" probablemente se solapa con otras señales de empleo formal y por eso LASSO la considera prescindible cuando hay penalización. En Ridge no se eliminan variables: los coeficientes se achican, pero no quedan exactamente en cero. En LASSO sí hay eliminación: el modelo deja coeficiente cero para variables completas como Edad, Categoría ocupacional, Horas Trabajadas, Cobertura médica: Obra Social y también Estado: Ocupado. Dicho de otra forma: "No paga" aporta información propia; "Obra social" muchas veces está contando lo mismo que ocupación formal/ingresos y por eso pierde relevancia cuando el modelo tiene que elegir.

En conjunto, esto muestra que LASSO está funcionando no solo como regularizador, sino también como un mecanismo de selección de variables, y que está produciendo un modelo más parsimonioso e interpretable cuando hay redundancia entre predictores.

B. Árboles

4. Metodología de Árboles de Decisión (CART)

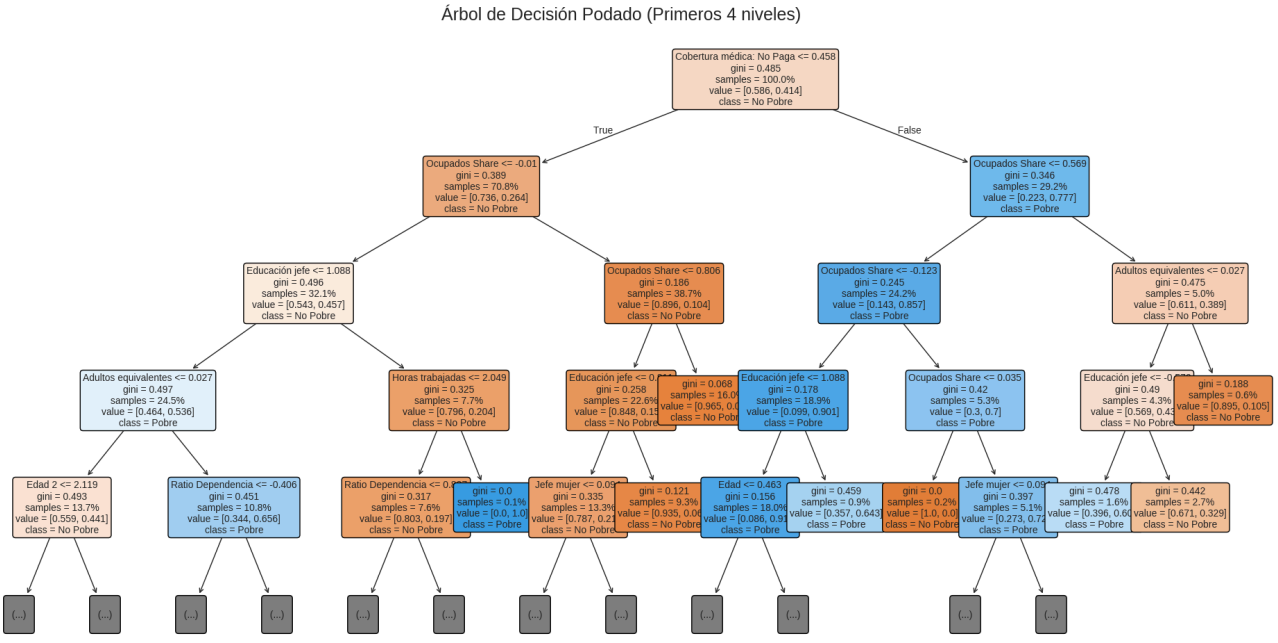
Figura 4. Error de clasificación vs. hiper parámetro de costo de complejidad del árbol.



El gráfico muestra una forma en "U" (o "V"), con una caída marcada del error cuando pasamos de valores muy bajos de ccp_alpha a valores intermedios. Esto sugiere que una poda moderada mejora la generalización: el árbol deja de capturar reglas muy específicas que no se sostienen fuera de muestra. El mínimo del error se alcanza alrededor de ccp_alpha ≈ 0.000923, por lo que ese valor aparece como el mejor balance entre complejidad y desempeño. A partir de ahí, el error vuelve a subir cuando ccp_alpha aumenta, lo que indica que el árbol se está simplificando demasiado y empieza a perder señal predictiva (subajuste).

5. Visualización e Interpretación del Árbol Podado

Figura 5. Árbol de decisión podado (primeros niveles).

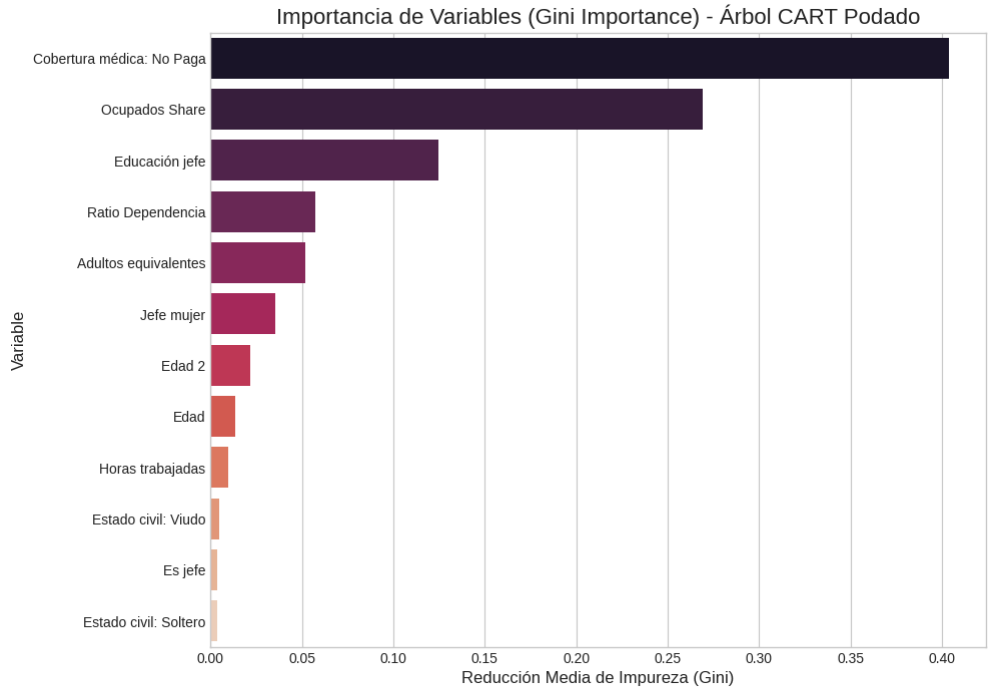


Dado el alfa óptimo se entrenó el árbol final. 12 niveles de profundidad y 57 nodos terminales hallados. Se visualizan los primeros 4 niveles de profundidad. La visualización del árbol da cuenta de la jerarquía de decisiones que definen la pobreza acorde a la muestra. La primera división es Cobertura médica: No paga. Este nodo raíz divide a la población en dos grupos principales: aquellos con cobertura formal (sea obra social o prepaga), aproximadamente el 71% de la muestra y son clasificados preliminarmente como "no pobres", el resto es identificado inicialmente como "pobres". Lo que esto pareciera dejar ver es que la formalidad laboral es el mayor protector frente a la vulnerabilidad económica; obviando así otras características sociodemográficas.

El modelo refina esta clasificación utilizando la proporción de miembros ocupados en el hogar (Ocupados Share), en esta segunda decisión aplica criterios dispares mostrando desigualdad estructural de los ingresos. El punto de corte para los formales está en el promedio poblacional (valor estandarizado de -0,01); mostrando que una intensidad laboral media es suficiente para sostenerse fuera de la pobreza. Pero, por el contrario, para el grupo sin cobertura médica, el modelo exige a un hogar un porcentaje de ocupados significativamente mayor, elevando el umbral a 0,56 desviaciones estándar por encima de la media; lo cual es relativamente coherente: dada la no protección formal, estos hogares tendrían que movilizar una tasa de ocupación excepcionalmente alta entre sus miembros para poder superar el umbral de pobreza. En la rama izquierda distingue no pobres por educación del jefe y vuelve a distinguir no pobres por valores extremadamente altos de Ocupados Share. En la rama derecha, categoriza como pobres aquellos por debajo de la media de ocupados.

En términos más generales, que el árbol óptimo tenga 12 niveles indica que la pobreza vendría a ser un fenómeno de "cola larga". Es decir, obviamente hay reglas generales, pero deben de existir grupos/clases específicos de pobreza que requieren condiciones particularmente puntuales para detectarse.

Figura 6. Importancia de variables (Gini)



"Feature Importance" mide cuánto reduce la impureza total cada variable ponderada por la cantidad de muestras que pasan por sus nodos.

¿Podría decirse que los coeficientes de las variables menos importantes son los que LASSO "achicó" a cero?

En general, diríamos que sí. Variables como si es o no jefe, Horas Trabajadas y Estado Civil tienen importancia casi nula en el gráfico de importancia del árbol. Son las mismas variables que LASSO redujo a cero. Ambos algoritmos, parecieran estar encontrando la misma estructura en los datos. Hasta nos es lógico de pensar: pareciera que estos algoritmos identifican que una vez que se conoce la proporción de ocupados del hogar (Ocupados Share), saber la cantidad exacta de horas trabajadas no aporta información relevante suficiente para justificar incluirla o darle tanto peso. Quizás esta sea solo nuestra hipótesis (al menos una de ellas) pero la convergencia metodológica debe de reforzar, como mínimo, su validez teórica.

C. Comparación entre métodos

6. Comparación Integral de Métodos y Desempeño

Tabla 2. Métricas: Resumen del desempeño en el set de prueba (X_test), utilizando un umbral de corte de probabilidad de 0.5.

Modelo	AUC-ROC	Accuracy	Error (1-Acc)	Recall (Pobre)	Precision (Pobre)	F1-Score
Logit (TP3)	0.8537	0.7774	0.2226	0.6523	0.7746	0.7082
KNN (TP3)	0.8204	0.7494	0.2506	0.6071	0.7408	0.6674
Logit-LASSO	0.8543	0.7813	0.2187	0.6560	0.7808	0.7130
Logit-Ridge	0.8539	0.7782	0.2218	0.6541	0.7751	0.7095
CART (Árbol)	0.8628	0.7907	0.2093	0.7538	0.7440	0.7488

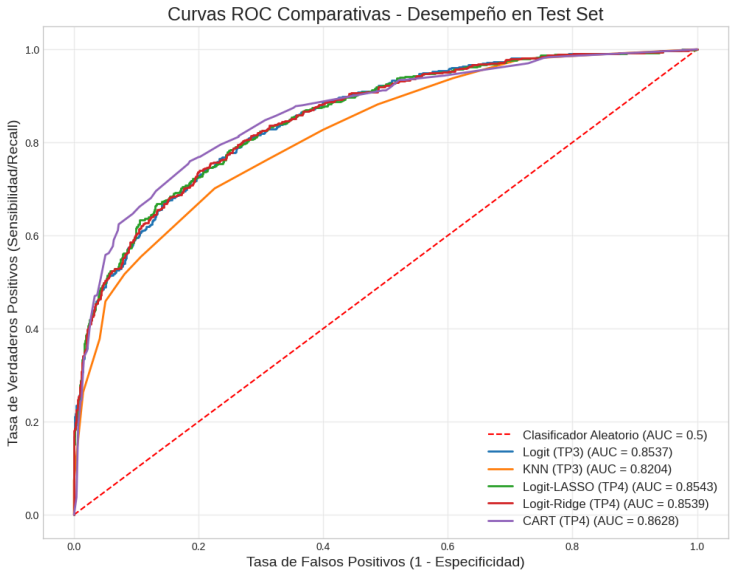
Podemos ver que los modelos logísticos tienen desempeños muy parecidos entre sí en términos de AUC. Es decir, la regularización no cambia de manera fuerte la capacidad de ranking del logit, pero sí lo "ordena" y, en el caso de LASSO, permite selección de variables sin perder desempeño. En contraste, KNN queda por debajo con AUC 0.8204, y el mejor AUC lo obtiene CART con 0.8628, lo que sugiere que el método no lineal captura mejor algunos patrones que los modelos lineales no representan tan bien.

Si miramos Accuracy y Error total, CART también queda primero: Accuracy 0.7907 (Error 1–accuracy = 0.2093), mientras que LASSO tiene accuracy 0.7813 (Error 0.2187), Ridge 0.7782 (Error 0.2218) y el Logit base 0.7774 (error 0.2226). La diferencia no es enorme, pero es consistente: CART reduce el Error total y mejora la Accuracy respecto del resto, mientras que la regularización mejora de forma leve o mantiene el desempeño del Logit base.

Sin embargo, para el problema de pobreza la diferencia más relevante no está en la Accuracy promedio, sino en cómo se comportan Precision y Recall de "Pobre". En la tabla vemos que CART tiene un Recall claramente más alto que los modelos logísticos. Esto significa que, con el umbral $p > 0.5$, el árbol detecta correctamente una proporción mayor de hogares pobres y reduce los falsos negativos. En cambio, en términos de Precision, LASSO y Ridge superan por poco al árbol, lo que muestra que el árbol incluye un poco

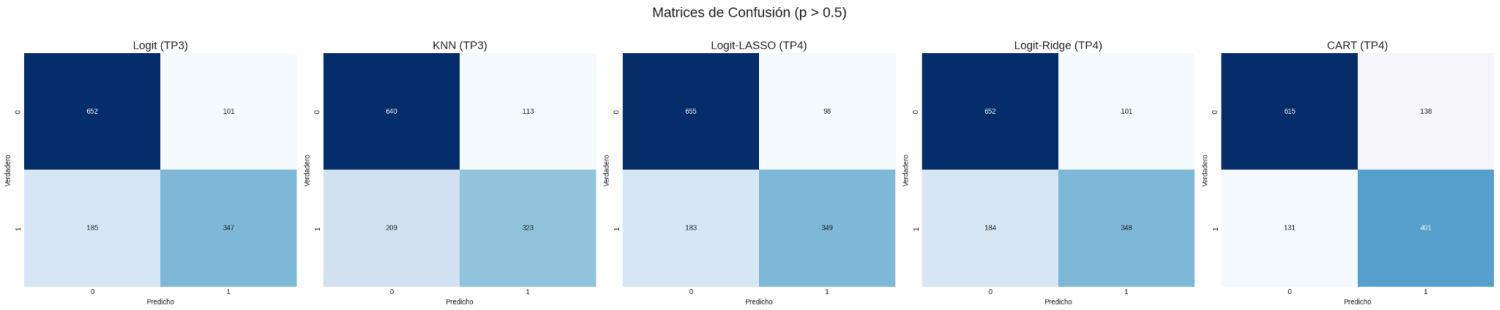
más de no pobres entre los que predice como pobres. Debido a este trade-off, es útil mirar el F1-score, que balancea Precision y Recall: CART obtiene el F1 más alto y los logit quedan alrededor de 0.708–0.713, mientras que KNN queda más abajo. En otras palabras, en CART mejora el equilibrio general porque sube mucho Recall sin que Precision caiga de forma exagerada.

Figura 7. Comparación de Curvas ROC



Las curvas ROC refuerzan la misma conclusión: la curva de CART aparece un poco por encima de las curvas logísticas y lejos de la diagonal aleatoria, consistente con su mayor AUC (0.8628). Los logits (sin penalidad, Ridge y LASSO) prácticamente se superponen, lo cual coincide con que sus AUC son casi idénticos. KNN queda por debajo, coherente con su menor AUC.

Figura 8. Matrices de confusión



Las matrices de confusión ayudan a entender por qué los modelos logit (incluyendo LASSO y Ridge) tienden a ser más "conservadores", con menos falsos positivos pero más falsos negativos, lo que baja el recall. El árbol, en cambio, recupera más verdaderos positivos (sube el recall), aunque suele pagar con algo menos de Precision. En nuestro caso, el balance termina siendo favorable para CART porque mejora Recall sin perder demasiado en Precision, y además reduce el error.

El árbol es intuitivo para explicar reglas pero es más difícil de resumir en un solo efecto promedio. La ventaja principal del método no lineal aparece en la performance: el árbol puede capturar interacciones y cortes por umbral que un logit lineal no captura de forma tan directa. En este set, sí se ve una ventaja en términos predictivos (AUC y recall).

7. ¿Cambió su respuesta con respecto a cuál es el "mejor" modelo para asignar recursos escasos a los más necesitados?

En el TP3, ante la elección entre Logit y KNN, la recomendación se inclinaba hacia el Logit por su estabilidad y capacidad explicativa. Sin embargo, este último trabajo cambia esa conclusión. Si el Ministerio de Capital Humano necesita identificar hogares vulnerables para dirigir recursos, el criterio de elección no debería ser solo "Accuracy", sino principalmente reducir falsos negativos, porque un falso negativo significa dejar afuera a un hogar pobre. En ese sentido, con umbral 0.5, el CART aparece como el candidato más fuerte: es el que logra el Recall más alto para pobres dentro de los modelos comparados. En nuestros resultados, el árbol identifica correctamente alrededor del 75.4% de los pobres, mientras que el mejor logit regularizado (LASSO) captura cerca del 65.6%. Dicho de forma simple: si usáramos Logit, estaríamos dejando afuera a aproximadamente un 10% más de la población objetivo que con CART, manteniendo el mismo umbral.

Dicho eso, también hay una idea importante: si el objetivo es maximizar la cobertura de pobres, una alternativa práctica con modelos logísticos es bajar el umbral de corte, porque eso sube Recall aunque baje precisión. En este TP, como comparamos todos con el umbral estándar, el CART ya ofrece esa cobertura superior sin tocar el umbral. Además, aunque el árbol tiene una precisión un poco menor (aprox. 0.744 vs 0.780 en LASSO, o sea que incluye un poco más de no pobres), su desempeño global sigue siendo mejor medido con F1. Por eso, si lo que más pesa es no dejar afuera a hogares pobres, es razonable elegir CART para asignar recursos. La decisión final depende del objetivo: si priorizamos interpretabilidad "económica" y simplicidad, logit/LASSO es muy defendible. Pero si priorizamos detectar más pobres con el mismo umbral, CART muestra una ventaja.