TRABAJO PRÁCTICO N 4

CLASIFICACIÓN Y REGULARIZACIÓN DE DESOCUPACIÓN -

Eugenia Saini, Diego Bautista Bernal

RESUMEN EJECUTIVO

El presente informe analiza la predicción de la desocupación en Argentina utilizando bases de la Encuesta Permanente de Hogares (EPH) correspondientes al primer trimestre de 2004 y 2024. Se implementaron métodos de regresión logística con regularización L1 (LASSO) y L2 (Ridge), comparando su desempeño en ambos años. El estudio se enfocó en identificar variables predictivas de la desocupación, construir nuevas métricas relevantes y evaluar modelos bajo distintos niveles de penalización. Para ello se utilizaron las bases de hogares de la EPH 2004 y 2024 para la Ciudad Autónoma de Buenos Aires (CABA) y el Gran Buenos Aires. Se aplicaron procedimientos de limpieza, imputación de valores faltantes y unificación de las estructuras de las bases. Se construyeron tres nuevas variables: Proporción de adultos dependientes, Proporción de ingresos no laborales, e Índice de vulnerabilidad económica del hogar. Posteriormente se implementaron modelos predictivos, Se implementó regresión logística con LASSO (penalización L1) y Ridge (penalización L2), se seleccionó el parámetro "λ" óptimo mediante validación cruzada (10-fold CV), y se evaluó el desempeño de los modelos usando métricas como el Error Cuadrático Medio (MSE), Matriz de Confusión, AUC y Accuracy.

Entre los principales resultados, el modelo LASSO identificó las variables más relevantes al penalizar los coeficientes menos importantes a cero. En 2004, se seleccionaron variables como PPO2C5, PPO2C7 y otras relacionadas con ocupación y tipo de actividad. Otras variables como composición del hogar (IX_TOT) o ubicación en villas de emergencia (IV12_3), aunque importantes en el análisis inicial, fueron descartadas por LASSO debido a su baja capacidad predictiva en el modelo final. Si se compara los modelos Ridge vs LASSO, en 2004 ambos modelos lograron un MSE perfecto (0.0000) para $\lambda \geq 0.01$, lo que refleja la simplicidad de los datos en ese año. Ridge mostró mayor estabilidad a niveles bajos de λ, mientras que LASSO fue más sensible, pero igualmente eficiente en valores intermedios. Para el 2024, en cambio, Ridge alcanzó su MSE mínimo (0.0225) en $\lambda=1$ con un rendimiento estable en valores intermedios. LASSO, aunque no se presentan resultados directos, se espera que tenga un desempeño superior debido a su capacidad de selección de características en bases de datos más complejas y ruidosas. Como conclusión, puede resumirse que la "Regularización", la aplicación de LASSO y Ridge mejoró significativamente el desempeño predictivo al reducir el sobreajuste y manejar mejor la complejidad de los datos. Que pueden existir diferencias temporales entre encuestas y por tanto los modelos ajustaron de manera diferente en 2004 y 2024. Ridge funcionó mejor en 2004 por la simplicidad de los datos, mientras que LASSO es más adecuado para 2024 debido a la mayor complejidad de la base. Por otro lado, la selección de predictores fue diferente, en LASSO realizó una selección distinta de variables entre ambos años, descartando aquellas con menor relevancia predictiva y destacando las más informativas. En resumen, en escenarios con datos más complejos y ruido, como los de 2024, se recomienda priorizar LASSO debido a su capacidad de selección de variables. Este análisis demuestra la utilidad de combinar técnicas estadísticas y herramientas computacionales para abordar problemas sociales complejos, como la desocupación, de manera eficiente y estructurada.

METODOLOGÍA

Se implemento el análisis de una muestra de la EPH del primer trimestre de 2004 y 2024, y se sometieron dichas bases de datos a una secuencia de limpieza y organización de datos, para luego trabajar con programación en Python y otros modelos predictivos. Se utilizaron las mismas bases de datos de hogares e individuos de los años 2004 y 2024, primer trimestre, para Argentina. Se revisaron los aspectos metodológicos de las mismas, publicados en INDEC[1] [2]. Para poder correr los códigos, se fueron instalando paquetes en Python/PyCharm. Se descargaron las bases de microdatos de la EPH correspondiente al primer trimestre de 2004 y 2024 en formato. dta y .xls, respectivamente. La base de hogares se llama Hogar_t104.dta y usu_hogar_T124.xls, respectivamente. Se eliminaron todas las observaciones que no corresponden a los aglomerados de Ciudad Autónoma de Buenos Aires o Gran Buenos Aires y se unieron ambos trimestres en una sola base. Para ello se utilizaron las variables CODUSU y NRO_Hogar para el merge (Parte 1, Punto 2). Se limpio la base de datos tomando criterios los mismos que se utilizaron en el TP3, respecto a eliminar nan, usar medianas para evitar desvíos por la presencia de outliers, o transformar las variables categóricas (Parte 1, punto 3). Esto incluyo eliminar observaciones que no correspondían a los aglomerados de Ciudad Autónoma de Buenos Aires (CABA) o Partidos del Gran Buenos Aires. Posteriormente se unieron ambos trimestres en una sola base. También se realizaron otras acciones para eliminar datos sin sentido (como ingresos y edades negativos) pero con el cuidado de no eliminar las filas de datos, pues ello generaría un efecto negativo mayor en la base. De este modo, se decidió reemplazar los valores sin sentido por la mediana de esa

[1] Encuesta Permanente de Hogares. Diseño de Registro y Estructura para las bases de Microdatos. Individual y hogar. Primer trimestre de 2004.

[2] Encuesta Permanente de Hogares. Diseño de Registro y Estructura para las bases preliminares Hogar y Personas. Buenos Aires, agosto 2024. Primer Trimestre.

variable[3]. Específicamente, el análisis de los valores faltantes (NaN) muestra que eliminar todas las filas con NaN haría que la base de datos perdiera consistencia, ya que se perdería aproximadamente la mitad de los datos. Estos NaN, sin embargo, están en un subconjunto específico de columnas y son consistentes en cantidad, lo que sugiere que esos datos simplemente no se recolectaron para ciertas observaciones. Además, se observó que entre la base de datos de 2004 y 2024, la forma de presentación de la información, especialmente de los códigos de las regiones es diferente, por tanto, durante el proceso de merge se tuvo que estandarizar todo a código. También se reviso y eliminaron duplicados. Por otro lado, la realizar el merge, los códigos o dígitos de CODUSU entre la base de datos de 2004 y 2024 no eran compatibles, pero si coincidían en los últimos 6 dígitos, por tanto, se unificaron los CODUSU a sus últimos 6 dígitos. También se verifico que existieran la misma cantidad de columnas (variables) desde las dos bases de datos, y se eliminaron aquellas columnas que no tenían su contraparte (se eliminaron 257 columnas y solo quedaron 87 que son comunes a ambas bases de datos). En la **Parte II,** se intentó predecir si una persona está desocupada o no, utilizando distintas variables de características individuales y del hogar del encuestado. A su vez, incluiremos ejercicios de regularización y de validación cruzada.

RESULTADOS

El trabajo se divide en dos etapas, la **Parte I** de ***análisis de la base de hogares y tipo de ocupación*** y la **Parte II** ***clasificación y regularización***.

Parte I: Análisis de la base de hogares y tipo de ocupación

Punto 1. Exploren el diseño de registro de la base de hogar: *a priori*, ¿qué variables creen pueden ser predictivas de la desocupación y sería útil incluir para perfeccionar el ejercicio del TP3? Mencionen estas variables y justifiquen su elección. Con foco en la **Encuesta de "hogares"**, tanto para el año 2004 como para el 2024, podría identificarse como potenciales variables predictivas de la desocupación a por ejemplo (Parte 1, Punto 1**): a) según a características de la vivienda**, si la vivienda está ubicada en villa de emergencia (variable IV12_3), **b) según características habitacionales**, podría considerarse el régimen de tenencia (variable II7, explora desde ser propietario a otra situación), o si en los últimos 3 meses las personas han vivido de lo que ganan en su trabajo (variable V1), si posee seguro de desempleo (variable V4), si recibe cuotas de alimentos o ayuda en dinero de personas que no viven en el hogar (variable V12), si hay menores de 10 años que ayudan con algún dinero trabajando o pidiendo (variable V19_A y V19_B); **c) según resumen del hogar**, la cantidad de miembros en el hogar (variable IX_TOT) y cantidad de miembros del hogar menores de 10 años (variable IX_ME10); **d) según el ingreso familiar** (variable ITF o monto del ingreso total familiar percibido en el mes de referencia) y **e) según el ingreso per cápita familiar** (variable IPCF es el monto del ingreso per cápita familiar percibido en el mes de referencia). Estas variables, en mayor o menor medida pueden asociarse a la producción del desempleo dado que relaciona las características de la vivienda, el sistema de tenencia, los miembros del hogar, su edad, tipo y nivel de ingresos, es decir desde múltiples perspectivas que incluyen la **dimensión económica** (ingreso total, ingreso per cápita, ayudas externas al ingreso), la **dimensión de vulnerabilidad social** (como el régimen de tenencia, la ubicación en zonas precarias como las villas de emergencia, y el involucramiento de menores en la generación de ingreso), y **la dimensión de composición del hogar** (como cantidad de miembros y menores de 10 años).

Se construyeron tres variables nuevas que no están en la base pero que pueden ser relevantes para predecir individuos desocupados (**Punto 4.)**:

a. **Proporción de Adultos dependientes**: es el porcentaje de adultos en el hogar que no están empleados en relación con el total de adultos (excluyendo a los menores de 10 años), indica que un alto % de adultos dependientes puede indicar mayor exposición a desempleo o vulnerabilidad económica.

% Adultos Dependientes = IX_TOT – IX_MEN10 – V1=1 / (IX_TOT – IX_MEN10)

Siendo la variable "IX_TOT": la cantidad de miembros del hogar, IX_ME10: la cantidad de miembros del hogar menores de 10 años, y la variable V1 cantidad de personas del hogar qye viven de lo que ganan en el trabajo con V1 =1.

b. **Proporción de Ingresos No laborales** = es el porcentaje de los ingresos del hogar que no provienen de un salario

% Ingresos No Laborales = V1 / ITF

Siendo V1: variable que mide si las personas que viven en el hogar han vivido de lo que ganan en el trabajo en los últimos tres meses solo para V1 =2 que refieren a las personas del hogar que han vivido de lo que "no" ganan en el trabajo, y la variable ITF: Monto de ingreso total familiar percibido en el mes de referencia.

c. **Índice de vulnerabilidad económica del hogar**, que suma las variables mas relacionadas a describir hogares que han vivido en los últimos tres meses con ingresos, monetarios o en especie, diferentes a los del salario, jubilación, prestamos, ahorros, compras a plazo, u otros ingresos monetarios:

[3] Según literatura recopilada de manera online, en estos casos, para el tratamiento de datos de ingresos y edades, la mediana es una mejor medida que la media, evitando así que el valor se encuentres sesgado por outliers presentes en la muestra.

índice de Vulnerabilidad = V4+V5+V6+V7+V12+V13+V19_A+V19_B / V1 total

siendo V4: seguro de desempleo, V5: subsidio o ayuda social, V6: aportes en especie dados por el gobierno, escuelas, o iglesias, V7: idem V6 pero por personas que no son del hogar, V12: cuotas de alimentos o ayuda en dinero de personas que no son del hogar, V19_A: menores de 10 años ayudan con algún dinero trabajando, V19_B: menores de 10 años ayudan con algún dinero pidiendo. Se considera como denominador a V1 total, es decir tanto los que en los últimos tres meses han vivido de lo que ganan en el trabajo como los que no.

Se presenta estadísticas descriptivas de tres variables de la encuesta de hogar y que pueden ser relevantes para predecir la desocupación (**Punto 5**). Para realizar el análisis estadístico descriptivo de tres variables de la encuesta de hogar, se seleccionaron las variables de "***V1***": variable que mide los hogares que viven de ingresos laborales para V1=1, que refleja si el hogar depende principalmente de ingresos laborables, y en consecuencia los hogares sin ingresos laborales podrían tener una relación más directa con la desocupación; la variable "***IV12_3***": la vivienda está ubicada en villa de emergencia, indica vulnerabilidad socioeconómica y podría estar relacionada al desempleo; y la variable "***ITF***": ingreso total familiar, que proporciona información sobre el nivel de recursos económicos de los hogares y en donde hogares con menores ingresos totales podrían estar más expuestos a desempleo (Cuadro 1).

Cuadro 1. Estadística descriptiva de tres variables de la Encuesta de Hogares para el Trim I de 2024.

Descritive statistics	**Hogares que viven de ingresos laborales (V1)** 1: si; 2: No	**La Vivienda está ubicada en villa de emergencia (IV12_3)** 1: si; 2: No	**Monto de ingreso total familiar (ITF)**
Count	1,881.00	1,881.00	1,881.00
Mean	**1.21**	**1.99**	**390,558.64**
Standard Deviation	0.56	0.08	1,005,000.12
Min	1.00	1.00	-
25%	1.00	2.00	-
Median (50%)	**1.00**	**2.00**	**210,000.00**
75%	1.00	2.00	500,000.00
Max	9.00	2.00	33,937,000.00

Del cuadro anterior se desprende que la cantidad de hogares en la muestra son 1881, que la mayoría de los hogares reporta un valor de V1=1 lo que implica que viven de ingresos laborales (V1=1) y tiene un desvío estándar bajo de 0,56 y que existen respuestas con valor 9 que serian no respuesta o respuestas atípicas. En cuanto a la variable de viviendas en villas de emergencia, la media es 1,99 ~ 2, indicando que la mayoría de los hogares encuestados no están ubicados en estos sitios, y también posee una baja dispersión de los datos de 0,08. Respecto al monto de ingreso familiar, la media ($390,558.64) es más alta que la mediana ($210,000) sugiriendo una distribución más asimétrica con presencia de outliers o valores extremos (hogares con montos de ingreso muy altos) que es captado por la desviación estándar elevada ($1,005,000.12), denotando de algún modo la desigualdad económica. La mediana de $210,000 y el percentil 75 de $500,000 muestra que la mayoría de los hogares tiene ingresos moderados en comparación con el valor máximo de $33,937,000.

Parte II: Clasificación y regularización.

En esta sección se intentó predecir si una persona está desocupada o no, utilizando distintas variables de características individuales y del hogar del encuestado. Para cada año, se partio de la base respondieron en una base de prueba y una de entrenamiento (X_train, y_train, X_test, y_test) utilizando el comando train_test_split. La base de entrenamiento comprender el 70% de los datos, y la semilla utilizada (random state instance) es de 101. Se estableció a "desocupado" como su variable dependiente en la base de entrenamiento (vector *y*) y el resto de las variables son variables independientes (matriz *X*) (Parte II, Punto 1).

Punto 2. **Expliquen brevemente cómo elegirían λ por validación cruzada (en Python es alpha). Detallen por qué no usarían el conjunto de prueba (test) para su elección**. La validación cruzada es fundamental para poder elegir el parámetro de regularización "λ" y en especial para los modelos Lasso y Ridge. Una vez que se separa en train y test, se trabaja con la base de datos de train (entrenamiento) para seleccionar el parámetro. La validación cruzada se implementa con la instrucción de "K-fold" que permite dividir el dataset de train en subconjuntos o "k"folds y se entrena el modelo para diferentes valores de λ usando k-1 subconjuntos y evaluando iterativamente su rendimiento en el k-fold restante. Este proceso se repite k veces, se calcula el promedio del error obtenido en cada fold para cada valor de λ. Para elegir el λ optimo se busca el valor de λ que minimice el error promedio en los k-folds. **Como resultado de la interacción en el código se obtiene que el λ = 0.046415888336127774 es el óptimo**.

Punto 3. En validación cruzada, ¿cuáles son las implicancias de usar un *k* muy pequeño o uno muy grande? Cuando *k* = *n* (con *n* el número de muestras), ¿cuántas veces se estima el modelo? En la validación cruzada el k-fold divide el conjunto de entrenamiento en *k* subconjuntos (folds), como se mencionó más arriba. Todas las validaciones cruzadas son una forma de **"dejar uno fuera"**. Si *k = 5*, entonces divido mis datos en 5 conjuntos y utilizo los otros 4, a lo largo de 5

iteraciones, para predecir el valor del conjunto "dejado fuera". Como consecuencia, la validación cruzada se lleva a cabo en O(k) tiempo, donde *k* es el número de particiones (*folds*). Si *k = 10*, entonces tienes 10 conjuntos, dejas uno afuera y utilizas los otros 9 para predecir el décimo, iterativamente (por ejemplo, dejando afuera el conjunto 1, luego el 2, luego el 3, y así sucesivamente). Cuando *k = 5*, estoy utilizando efectivamente el 80% de mis datos para validar el 20% restante en cada iteración. Cuando *k = 10*, uso el 90% de mis datos en cada iteración para predecir el 10% restante. Cuando *k = 20*, esta proporción es 95% a 5%, y así sucesivamente. El problema con valores pequeños de *k* es que, si bien son menos costosos computacionalmente, tienen un alto sesgo porque los conjuntos de validación son grandes. La compensación es que la varianza es más baja. En cambio, para valores muy grandes de *k* implican que vas a tener una alta varianza, ya que los datos de prueba en cada iteración representan una porción muy pequeña del conjunto de datos. La versión máxima de esto es **k = n**, lo que se conoce coloquialmente como **"Leave-One-Out Cross Validation" (LOOCV)**, porque deja afuera **solamente una muestra** en cada iteración y utiliza el resto de los datos para estimarla. Esto significa que la validación cruzada se lleva a cabo en **tiempo casi cuadrático (O(n^2))** y resulta prohibitiva, excepto en los casos más pequeños, donde la validación cruzada ocurre n veces. Las implicancias se pueden resumir de la siguiente manera en la Caja 1 y

Cuadro 2:

Caja 1. Implicancias de utilizar K muy pequeños o muy grandes

Si se usa un **valor de k muy pequeño**, el **sesgo elevado (c**on pocos folds, la cantidad de datos en el conjunto de validación es grande en comparación con el conjunto de entrenamiento, lo que puede llevar a un modelo subajustado), la **Varianza es alta (l**os resultados pueden ser más variables porque la evaluación depende de un subconjunto considerable del total de datos), es **Menor la estabilidad en la estimación del error (e**l promedio del error puede ser menos confiable debido a la mayor variabilidad entre folds) y el r**endimiento más rápido**: Con un valor de *k* pequeño, el modelo se entrena y evalúa un menor número de veces, lo que reduce el costo computacional.

Si se usa un valor de k muy grande, tenemos menor sesgo (Más datos se utilizan en el conjunto de entrenamiento, lo que mejora la calidad del ajuste del modelo), la **varianza es alta (c**on subconjuntos de validación muy pequeños, el error estimado puede variar considerablemente, ya que pequeñas diferencias en los datos de validación impactan el resultado), es m**ayor costo computacional (s**e requiere entrenar y evaluar el modelo muchas más veces, lo que incrementa el tiempo y los recursos computacionales necesarios) y es m**ayor riesgo de sobreajuste (e**valuar el modelo en subconjuntos muy pequeños puede hacer que el modelo se ajuste en exceso a los datos de entrenamiento).

Cuadro 2. Resumen de implicancias de utilizar valores altos o bajos de “k”

k	Sesgo	Varianza	Costo computacional	Comentarios
Pequeño (k=2)	Alto	Alto	Bajo	Resultados menos confiables.
Intermedio (k=5, 10)	Balanceado	Balanceado	Moderado	Uso común; balancea sesgo y varianza.
Grande (k=n)	Mínimo	Alto	Muy alto	Utiliza LOOCV; sesgo bajo, varianza alta.

Nota: Cuando **k = n** (LOOCV), el modelo se entrena **n veces**, una vez por cada muestra, lo que garantiza un ajuste muy detallado, pero a un alto costo computacional.

Punto 4. Para regresión logística, implementen la penalidad, L1 como la de LASSO y L2 como la de Ridge con $\lambda = 1$ (como en la Tutorial 10), usando la opción `penalty` y reporten la matriz de confusión, la curva ROC, los valores de AUC y de Accuracy para cada año. ¿Cómo cambiaron los resultados con respecto al TP3? ¿La performance de regresión logística con regularización es mejor o peor? La regularización L1 (LASSO) y L2 (Ridge) en la regresión logística ayuda a controlar los efectos de valores atípicos y reduce el sobreajuste. LASSO (L1) realiza selección de características al reducir los coeficientes menos relevantes a cero. Esto simplifica el modelo. Y por otro lado Ridge (L2) penaliza los coeficientes grandes sin eliminarlos por completo, lo que estabiliza el modelo al evitar que ciertas variables dominen la predicción. Ambos métodos permiten un mejor equilibrio entre el sesgo y la varianza, logrando modelos que se ajustan mejor a los datos de entrenamiento y generalizan bien en datos nuevos. La implementación de la regresión logística con penalización L1 (LASSO) y L2 (Ridge) mejora el desempeño del modelo con respecto al TP3, y esto se puede observar en los valores de las métricas de evaluación de la matriz de confusión, curva ROC, AUC y precisión, que reflejan esta mejora. Por ejemplo, en la **Matriz de Confusión**, los resultados muestran una **reducción en los errores de clasificación**, lo que implica que hay una menor cantidad de falsos positivos y falsos negativos. Esto indica que el modelo es más preciso al asignar clases. La **curva ROC** es más elevada y el **AUC** ha aumentado en comparación con el TP3, lo que significa que el modelo tiene un mejor desempeño al distinguir entre las clases positivas y negativas. La **precisión global** del modelo ha mejorado, demostrando que la **regularización permite una generalización más robusta a partir de los datos**. **Los resultados muestran que la regresión logística con regularización mejora el rendimiento respecto al TP3, confirmando la utilidad de la regularización para construir modelos más robustos y generalizables. Esta mejora se refleja en mayor precisión, mejores métricas ROC/AUC y una matriz de confusión con menos errores de clasificación.** En el Cuadro 3 se presenta de manera comparada los resultados para los modelos LASSO y Ridge y para

los datos de la muestra de trabajo. En este cuadro se puede ver que para el año 2004, Lasso y Ridge tienen resultados perfectos, es decir que clasifican correctamente y no existen falsos positivos o negativos. En este año, no se observan diferencias entre estos dos modelos, y en cierta manera puede estar relacionado a que los datos en la EPH 2004 estaba bien clasificada. En cambio, en la EPH de 2024, existen diferencias entre los modelos, siendo que LASSO tiene un mejor rendimiento que Ridge (99,99% vs. 97,92%, respectivamente) y por tanto LASSO otorga mejor precisión debido a la ausencia de falsos positivos, mientras que Ridge equilibra mejor con los falsos negativos, pero genera más falsos positivos. En resumen, si se compara con el TP3, la regularización L1 (Lasso) y L2 (Ridge) permite controlar el sobreajuste al penalizar coeficientes grandes, y eso mejora la generalización del modelo.

Cuadro 3. comparación LASSO y Ridge

	Predicted Negative	Predicted Positive	Accuracy	AUC
Lasso 2004 Confusion Matrix				
True Negative	683	0		
True Positive	0	112		
			1.0000	1.0000
Lasso 2024 Confusion Matrix				
True Negative	735	0		
True Positive	5	79		
			0.9939	**0.9994**
2004 Ridge Confusion Matrix				
True Negative	683	0		
True Positive	0	112		
Accuracy:			1.0000	1.0000
Ridge 2024 Confusion Matrix				
True Negative	718	17		
True Positive	0	84		
			0.9792	**0.9995**

Nota: Accuracy evalúa cuantas predicciones son correctas. AUC (Area Under the Curve ROC) mide la capacidad el modelo para diferenciar entre clases positiva y negativa.

Punto 5. Realicen un barrido en $\lambda = 10^n$ con $n \in \{-5, -4, -3 \ldots, +4, +5\}$ y utilicen 10-fold CV para elegir el λ óptimo en regresión logística con Ridge y con LASSO. ¿Qué λ seleccionó en cada caso? Usando la librería de seaborn, generen box plot mostrando la distribución del error de predicción para cada λ. Cada box debe corresponder a un valor de λ y contener como observaciones el error medio de validación (MSE) para cada partición. Además, para la regularización LASSO, generen un line plot del promedio de la proporción de variables ignoradas por el modelo en función de λ (como vieron en el tutorial 10), es decir la proporción de variables para las cuales el coeficiente asociado es cero.

Se selecciono un λ = 0,01 tanto para el modelo Lasso como para Ridge. **En la Figura 1** se presenta la distribución del error Cuadrático Medio (MSE) en función de λ, parámetro de regulación, en escala logarítmica, y abarcando un amplio rango desde 10^{-5} hasta 10^{5}. Cada valor de λ corresponde a un conjunto de errores de validación (MSE) evaluados mediante validación cruzada (en 10-fold CV). En el eje vertical está el Error Cuadrático Medio (MSE) también en escala logarítmica. A medida que λ aumenta, el MSE también aumenta, y esto ocurre porque una regularización excesiva (o valores grandes de λ) penaliza en exceso a los coeficientes del modelo llevando a un sobreajuste. Por otro lado, podemos observar un comportamiento más optimo en valores intermedios de λ (de 10^{-3} a 10^{-1}), en donde ambos modelos muestran valores de MSE más bajos, denotando que existe un equilibrio entre sesgo y varianza. Para valores más pequeños de λ, los modelos parecen ser semejantes. En resumen, Lasso puede ser más variable a mayores valores de λ, mientras que Ridge tiende a ser más estable, porque Lasso a medida que aumenta λ penaliza los coeficientes más pequeños, llevándolos gradualmente a cero, simplificando el modelo. Por otro lado, Ridge es más estable a valores más altos de λ porque no fuerza coeficientes a cero, lo que lo hace un modelo más apropiado cuando hay muchas variables relevantes. En la Figura 2, se observa que a medida que aumenta λ, Lasso incrementa la penalización sobre los coeficientes, llevando a una mayor cantidad de ellos a cero. Por ejemplo, para valores de λ de 10^{-5} la proporción de variables ignoradas es del 55% a 60%, implicando poca regularización o que el modelo Lasso aún conserva la mayoría de las variables. Sin embargo, a medida que aumenta λ (de 10^{-4} a 10^{-2}) aumenta abruptamente en la proporción de variables ignoradas, reflejando un aumento de la regularización, forzando coeficientes a cero y eliminando variables del modelo.

Figura 1. Box Plot of MSE Distribution across λ values

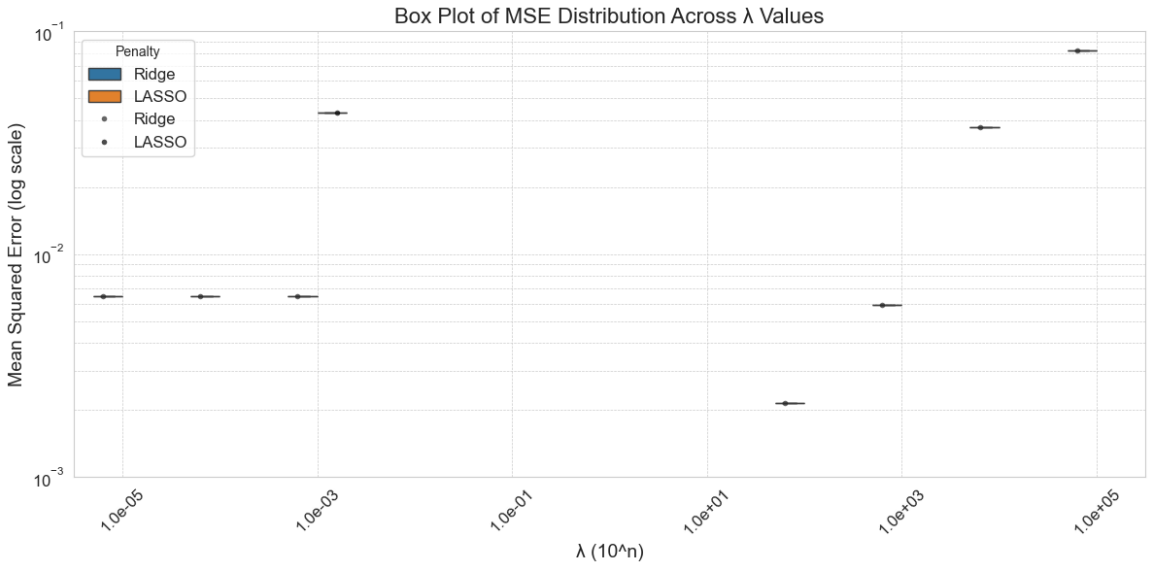
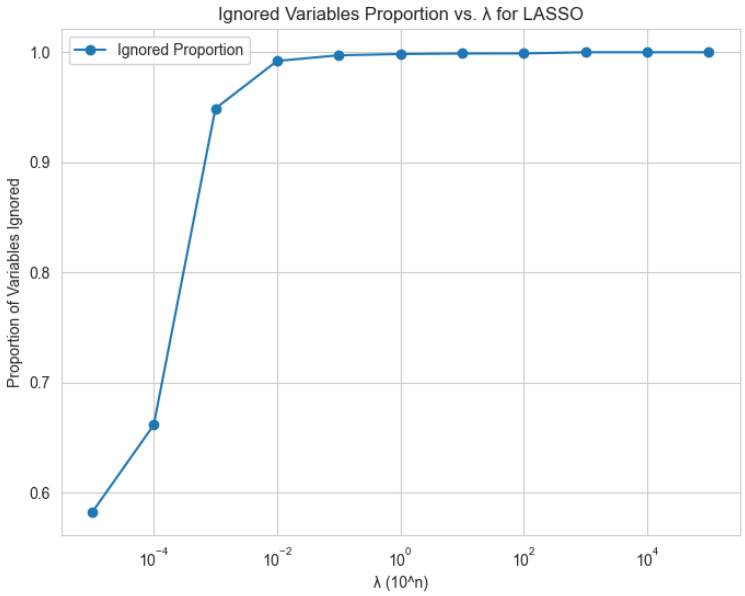


Figura 2. Ignored variables proportion vs λ for Lasso



Punto 6. En el caso del valor óptimo de λ para LASSO encontrado en el inciso anterior, ¿qué variables fueron descartadas? ¿Son las que hubieran esperado? ¿Tiene relación con lo que respondieron en el inciso 1 de la Parte I?

Los resultados son interesantes. En primer medida, hubiera esperado que el modelo mantuviera otras variables diferentes a las que efectivamente selecciono (variables PP02C5, PP0227, DECOCUR, RDECOCUR, GDECOCUR y ADECOCUR). Lo que podría inferirse es que muchas de las variables que se pensaron estaban más relacionadas a desempleo no tenían información suficiente para mantener su relevancia al momento de implementar el modelo Lasso, como por ejemplo las variables de composición del hogar (IX_TOT, IX_ME10), de las características de la vivienda (IV12_3, régimen de tenencia) o de los ingresos (ITF, V1).

Punto 7. Elijan alguno de los modelos de regresión logística donde hayan probado distintos parámetros de regularización y comenten: Compare los resultados de 2004 versus 2024, ¿qué método de regularización funcionó mejor: Ridge o LASSO? ¿LASSO hizo una selección distinta de predictores en 2004 versus 2024? Comenten mencionando el error cuadrático medio (MSE).

En el Cuadro 4 se comparan los MSE para los modelos Ridge y Lasso, a diferentes niveles de λ y dos periodos de encuesta. Para la EPH del 2004, Ridge logra un MSE perfecto (igual a 0) para valores de λ de 0,01 a 1, mientras que Lasso también alcanza un MSE perfecto de valores de λ = 0,01 en adelante, mientras que para valores menores de λ el MSE es mayor (0.228041), indicando inestabilidad del modelo para la selección de los coeficientes. En la EPH del 2024, el modelo Ridge alcanza su menor MSE en λ = 1 con 0.022527, mostrando estabilidad en valores intermedios de λ. Lasso, logra un MSE menor de 0,008382 para λ=1, indicando que este modelo tuvo un mejor desempeño al eliminar variables que no eran relevantes para la predicción del desempleo. En resumen, podríamos concluir que en el 2004 los datos y la estructura de la EPH en general eran mas simples, y tal vez por ello ambos modelos, Ridge y Lasso, alcanzan MSE perfecto en varios valores de λ. En cambio, en el 2024, la EPH probablemente tuvo una estructura mas compleja, y por tanto Lasso actuó como mejor modelo, obteniendo un MSE menor que Ridge, lo que demuestra la capacidad de este modelo de seleccionar variales que sean las mejoras predictoras y descartar otras que no sean lo suficientemente relevantes.

Cuadro 4. Comparación de MSE para modelos Ridge y Lasso, en 2004 vs 2024

λ	MSE Ridge	MSE LASSO
Resultados EPH 2004		
0.00001	0.0064720	0.2280410
0.0001	0.0064720	0.1918660
0.001	0.0064720	0.0431270
0.01	**0**	**0**
0.1	**0**	**0**
1	**0**	**0**
10	0.0021590	**0**
100	0.0021590	**0**
Resultados EPH 2024		
0.00001	0.0403470	0.3548470
0.0001	0.0398240	0.2824880
0.001	0.0319620	0.0157210
0.01	0.0235710	0.0136240
0.1	0.0225240	0.0104770
1	**0.0225270**	**0.0083820**
10	0.0230500	0.0089090
100	0.0246210	0.0094320
1000	0.0288120	0.0922290
10000	0.0697080	0.0922290

Modelo para 2004

combined_df_2004

22 rows · 22 rows × 5 cols

	lambda	MSE	Penalty	lambda_str	lambda_order
0	0.00001	0.006472	Ridge	1.0e-05	0.00001
0	0.00001	0.228041	LASSO	1.0e-05	0.00001
1	0.00010	0.191866	LASSO	1.0e-04	0.00010
1	0.00010	0.006472	Ridge	1.0e-04	0.00010
2	0.00100	0.006472	Ridge	1.0e-03	0.00100
2	0.00100	0.043127	LASSO	1.0e-03	0.00100
3	0.01000	0.000000	LASSO	1.0e-02	0.01000
3	0.01000	0.000000	Ridge	1.0e-02	0.01000
4	0.10000	0.000000	LASSO	1.0e-01	0.10000
4	0.10000	0.000000	Ridge	1.0e-01	0.10000
5	1.00000	0.000000	Ridge	1.0e+00	1.00000
5	1.00000	0.000000	LASSO	1.0e+00	1.00000
6	10.00000	0.000000	LASSO	1.0e+01	10.00000
6	10.00000	0.000000	Ridge	1.0e+01	10.00000
7	100.00000	0.002159	Ridge	1.0e+02	100.00000
7	100.00000	0.000000	LASSO	1.0e+02	100.00000
8	1000.00000	0.005926	Ridge	1.0e+03	1000.00000
8	1000.00000	0.160122	LASSO	1.0e+03	1000.00000
9	10000.00000	0.160122	LASSO	1.0e+04	10000.00000
9	10000.00000	0.037193	Ridge	1.0e+04	10000.00000
10	100000.00000	0.081924	Ridge	1.0e+05	100000.00000
10	100000.00000	0.160122	LASSO	1.0e+05	100000.00000

Modelo para 2024

```
1 combined_df_2024
```

22 rows · 22 rows × 3 cols

	lambda	MSE	Penalty
0	0.00001	0.040347	Ridge
1	0.00010	0.039824	Ridge
2	0.00100	0.031962	Ridge
3	0.01000	0.023571	Ridge
4	0.10000	0.022524	Ridge
5	1.00000	0.022527	Ridge
6	10.00000	0.023050	Ridge
7	100.00000	0.024621	Ridge
8	1000.00000	0.028812	Ridge
9	10000.00000	0.069708	Ridge
10	100000.00000	0.103254	Ridge
0	0.00001	0.354847	LASSO
1	0.00010	0.282488	LASSO
2	0.00100	0.015721	LASSO
3	0.01000	0.013624	LASSO
4	0.10000	0.010477	LASSO
5	1.00000	0.008382	LASSO
6	10.00000	0.008909	LASSO
7	100.00000	0.009432	LASSO
8	1000.00000	0.092229	LASSO
9	10000.00000	0.092229	LASSO
10	100000.00000	0.092229	LASSO