
**Trabajo Práctico N4**

Ciencias de Datos

2 Cuatrimestre 2024

Profesores: Ignacio Spiousas y Maria Noelia Romero

Alumnos:

Ariela Mirmelstein

Leandro Garbulsky

Analia Weibel Perl

**Correcciones TP3**

Antes de comenzar, corregimos los drops de aglomerado de cada uno de los df (2004, 2024), para que efectivamente en ambos permanezcan Ciudad de Buenos Aires/33 y Gran Buenos Aires/32. Además, reemplazamos Ciudad de Buenos Aires/33 por "CABA", y Gran Buenos Aires/32 por "GBA".

**PARTE 1**

**1.** La base de *hogar* tiene diversas variables que pueden ser predictivas de la desocupación. Dentro de ellas las podemos separar diferentes categorías:

1. Variables relacionado con el entorno socioeconómico del hogar: región, mas_500 (tamaño de aglomerado), pondera. Las tasas de desocupación pueden variar significativamente en base a la región, tamaño y observación en la población total.
2. Variables sobre características de la vivienda:IV1 (tipo de vivienda), IV6 (acceso al agua dentro de la vivienda), IV6(acceso al agua dentro de la vivienda) y IV12_3 (vivienda en la villa de emergencia). Ya que son indicadores socioeconómicos que están asociados a contar con riesgos de desocupación.
3. Variables relacionadas con la economía del hogar: V1 (ingresos laborales), V5(subsidios). Hogares que no dependen de ingresos pueden tener una proporción más alta de desocupación.
4. Variables sobre composición y condiciones del hogar: IX_TOT (tamaño del hogar), IX_MEN10 (cantidad de menores de 10 años) y II7 (régimen de tenencias).

**2 y 3 . Manejo de base y limpieza.**

Una vez descargadas las bases para cada año, se decidió pasar todo a una misma tipografía minúscula para asegurar la uniformidad en los campos de texto. Luego, se filtraron únicamente las observaciones correspondientes a los aglomerados solicitados, identificados por los códigos **32** (Gran Buenos Aires) y **33** (Ciudad Autónoma de Buenos Aires, CABA).

Posteriormente, se realizó un control de tipos de datos claves para evitar errores durante el análisis con las variables ‘codusu’ y ‘nro_hogar’ que se utilizaran en el código. Se aseguró que el primero sea un sting y el segundo un número entero.

Para garantizar una correcta integración de las bases correspondientes a cada año, se identificaron las diferencias en las columnas, sacando la columna *‘idimph’* que no está en el año 2024, y en hogar_2024 se identificó la columna *‘pondih’* que no está en 2004. Dado que estas columnas no presentan información relevante para el posterior análisis, se decidió borrarlas con la función *drop()*.

A su vez, se agregó una columna llamada *‘anio’* para poder identificar y categorizar las variables en 2004 y 2024. A partir de ello, se concatenan las dos bases de Hogar para unificar los años.

Luego para identificar si hay datos faltantes en las columnas se utilizó la función isnull().sum() y se filtró las columnas que tienen al menos un dato faltante dando como resultado 10 columnas con este tipo de inconsistencia.

Después, para consolidar la información de los individuos y sus hogares en un único conjunto de datos, se realizó una operación de merge entre las bases *df_conjunto* (datos

individuales que se había utilizado en el TP3) y *df_hogares* (datos de los hogares de cada año). A partir de este momento, se comenzó a utilizar la función *df_merged.*

En base al nuevo dataframe, se identificaron columnas duplicadas generadas debido a la existencia de nombres que coincidan en ambos conjuntos. Estas duplicaciones se manifestaron en el código con sufijos automáticos añadidos por pandas, como "_x" y "_y". Con la aparición de "_x" como "_y", se conservó "_x" por defecto y se eliminó la versión "_y" , y si solo existía una de las dos versiones el sufijo fue eliminado para restaurar el nombre original de la columna. Se chequeo el tipo de categoría de las variables 'nivel_ed' y 'realizada'.

Dado que varias variables en el conjunto de datos consolidado (df_merged) son de tipo categórico, se realizó un análisis para identificar los valores únicos presentes en este tipo de columnas y detectar así las inconsistencias como mezcla recurrente de texto y números. El listado de columnas categóricas analizadas fueron: "ch04": Sexo; "ch07": Estado civil; "ch08": Cobertura médica; "nivel_ed": Nivel educativo; "estado":Condición de actividad laboral; "cat_inac": Categoría de inactividad laboral; "realizada": Estado de realización de la entrevista; "iv1": Tipo de vivienda; "ii7": Régimen de tenencia de la vivienda; "v5": Si recibe ayuda alimentaria o mercadería.

Con el objetivo de unificar y simplificar el formato de las variables categóricas, se implementó un proceso de conversión basado en un diccionario predefinido de mapeos. Se reemplaza las etiquetas textuales por valores numéricos, para facilitar el análisis posterior.

Con las transformaciones y columnas listas, la limpieza de datos faltantes o valores no válidos fue fundamental para garantizar la calidad de los resultados. Primero, se seleccionaron variables que fueron identificadas como relevantes en las bases debido a su importancia a la hora de predecir desocupación. Estas variables representan distintos aspectos demográficos, sociales, económicos de interés: 'anio', 'ch04', 'ch06', 'ch07', 'ch08', 'nivel_ed', 'estado', 'cat_inac', 'ipcf', 'codusu', 'nro_hogar', 'codusu', 'realizada, 'iv1', 'iv2', 'ii7', 'v5', 'ii1', 'ix_tot'. En estas columnas no se registraron Nans.

Luego, se convirtieron los valores negativos que no tenían sentido a Nan y luego los imputamos con la mediana de edad. Para ello se controló las variables ch06 y nivel_ed.
Cantidad de valores NaN en 'ch06': 186 / Cantidad de valores NaN en 'ch06' después de la imputación: 0

Posteriormente se hizo lo mismo con *ingresos*, se aseguró que los valores negativos o ceros sean eliminados y se imputaron las observaciones faltantes. Se controló por 'ipcf' (Ingreso per cápita) y 'itf' (Ingreso total familiar) dado su carácter informativo. Cantidad de valores NaN en 'ipcf': 3134 / Cantidad de valores NaN en 'itf': 3134.

Para concluir la limpieza, se observó si el df_merged tenía valores no válidos. Las columnas involucradas son: iv2, ii1, ix_tot y ipcf ya que no pueden ser negativas debido a lo que representa,

Con la función *shape()* verificamos cómo queda la base luego de la limpieza. Cantidad de datos después de la limpieza: 11564 filas, 238 columnas.

Por último, se seleccionó *'realizada'* para volver a verificar si había diferencia sobre el tipo de objeto del df. Todos los resultados dieron valores numéricos.

Para corroborar la limpieza de datos faltantes volvimos a seleccionar las variables y a certificar que solo queden observaciones válidas.

## 4. Construcción de variables.

Para mejorar la capacidad predictiva del modelo de desocupación, construimos tres nuevas variables derivadas de las columnas ya existentes del merged.

Variable 1: Estabilidad de la vivienda ('estabilidad_vivienda'), para la construcción de esta variable tuvimos en cuenta los distintos valores que toma tipo de vivienda (iv1) y se les asignó la característica de estable o inestable. 1 y 2 (Casa y Departamento) se categorizan como "estable", mientras que 3, 4, y 5 (Inquilinato, Hotel/pensión, Local no construido para vivienda) se categorizan como "inestable".

Variable 2: Proporción de miembros con cobertura pública de salud ('cobertura_hogar'), esta nueva variable calcula para cada hogar identificado por las columnas codusu y numero hogar, la proporción de miembros del hogar que tienen cobertura médica pública.

"ch08" informa el tipo de cobertura médica de cada miembro del hogar, con la condición (x == 3) se verifica si el valor de ch08 es igual a 3, lo que indica que tienen cobertura médica pública

Variable 3: Nivel educativo máximo del hogar ('ed_max_hogar'), esta nueva variable calcula el nivel máximo de educación alcanzado por cualquier miembro del hogar

## 5. Estadísticas descriptivas.

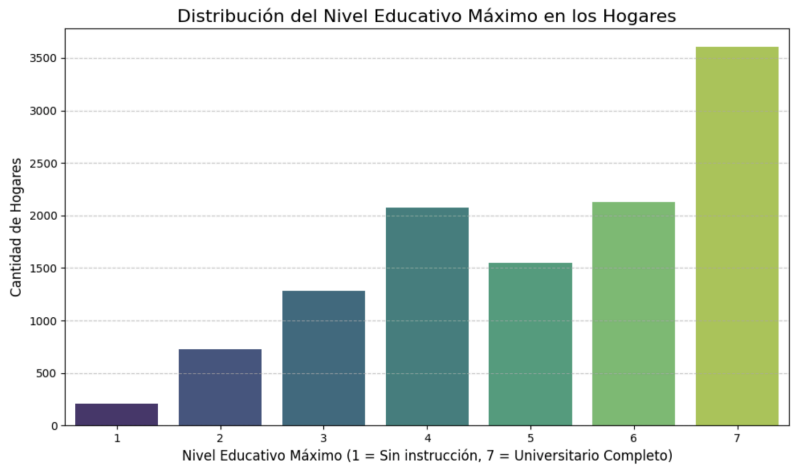
Para presentar estadísticas descriptivas de tres variables relevantes para predecir la desocupación, utilizamos la función *describe*() y seleccionamos 3 variables que reflejan condiciones que influyen en la probabilidad de estar desocupado y filtramos por año. Estas variables son: 'ipcf', 'ed_max_hogar', 'cobertura_hogar'.

Estadísticas descriptivas por año:

| | ipcf | | | | | | | | ed_max_hogar | | | | | cobertura_hogar | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | count | mean | std | min | 25% | 50% | 75% | max | count | mean | ... | 75% | max | count | mean | std | min | 25% | 50% | 75% | max |
| anio | | | | | | | | | | | ... | | | | | | | | | | |
| 2004 | 7552.0 | 367.070842 | 746.613018 | 8.333333 | 123.2 | 226.666667 | 443.75 | 54000.00 | 7552.0 | 5.157707 | ... | 7.0 | 7.0 | 7552.0 | 0.006224 | 0.066924 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 |
| 2024 | 4012.0 | 281321.811787 | 475749.948315 | 3675.000000 | 100000.0 | 175000.000000 | 324000.00 | 11312333.33 | 4012.0 | 5.125872 | ... | 6.0 | 7.0 | 4012.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

Monto del ingreso per cápita familiar. Se puede ver una diferencia en las medias y valores máximos de los años. Sugiere que los valores de *ipcf* en 2024 son mucho más altos y con una variabilidad mayor. Esto podría indicar un cambio en la distribución de la pobreza, con un aumento en las diferencias entre los hogares.

Nivel educativo máximo del hogar**.** Los valores de *ed_max_hogar* muestran que el nivel educativo máximo de los hogares permaneció estable a lo largo del tiempo con pequeñas diferencias.




En este gráfico podemos ver la distribución de los hogares según el nivel educativo máximo de sus miembros, y una clara tendencia hacia niveles educativos más altos, con un pico notable en el nivel 7. El hecho de que el tercer cuartil se encuentre en un nivel educativo relativamente alto coincide con la observación del histograma.

Cobertura médica del hogar. En 2024, *cobertura_hogar* parece ser uniforme con todos los hogares sin cobertura. En 2004, hubo una pequeña proporción de hogares con cobertura médica (aproximadamente 0.6%).

**PARTE 2**

1. El objetivo de este inciso es preparar los datos para predecir la condición de desocupación de los encuestados (variable dependiente desocupado) mediante la partición de la base de datos en conjuntos de entrenamiento y prueba, para los años 2004 y 2024. Además, se realiza el preprocesamiento de las variables independientes necesarias para los modelos de clasificación. La metodología surgió siguiendo estos pasos:

a. **Filtrado por Año:**
   La base de datos fue dividida en dos subconjuntos, uno para el año 2004 (datos_04) y otro para 2024 (datos_24), utilizando la columna anio como indicador.

b. **Creación de la Variable Dependiente** (desocupado):
   Se definió desocupado como una variable binaria:
   - Valor 1: Personas clasificadas como desocupadas (estado = 2).
   - Valor 0: Personas ocupadas, inactivas o menores de edad.

c. **Selección de Variables Independientes:**
   Se utilizaron variables relacionadas con características individuales y del hogar, tales como nivel educativo, ingresos, estabilidad en la vivienda y cobertura de salud. Las variables seleccionadas son:
   - **Características individuales**: ch04, ch06, ch07, ch08, nivel_ed, cat_inac.
   - **Características del hogar:** ipcf, iv1, iv2, ii7, v5, ii1, ix_tot, estabilidad_vivienda, cobertura_hogar, ed_max_hogar.

d. **Partición en Entrenamiento y Prueba:**
   Se utilizó el comando train_test_split para dividir los datos en conjuntos de

entrenamiento (70%) y prueba (30%) con una semilla de aleatoriedad fija (random_state=101) para garantizar la reproducibilidad.

e. **Preparación de las Matrices de Datos:**
Se añadió una columna de intercepto (Intercepto) en las matrices de entrenamiento y prueba para incluir un término constante en los modelos.
Las variables categóricas fueron convertidas a dummies con la opción drop_firs t= True para evitar colinealidad.

El preprocesamiento de datos permitió generar conjuntos de entrenamiento y prueba para 2004 y 2024 con las características necesarias para el análisis de clasificación. La creación de dummies y la normalización de las variables aseguraron que las matrices estuvieran listas para ser utilizadas en modelos de regresión logística con regularización.

**2. Elección de λ para validación cruzada.**

Para elegir el valor de λ, usaríamos validación cruzada. Este método consiste en dividir el conjunto de entrenamiento en varias partes (llamadas folds) y probar diferentes valores de λ para ver cuál funciona mejor. Por ejemplo, podríamos dividir el conjunto en 10 partes y, en cada iteración, usar 9 de ellas para entrenar el modelo y la parte restante para validarlo. Esto se repite hasta que todas las partes hayan sido usadas para validar.

La idea es probar distintos valores de λ y calcular un promedio del error obtenido en todas las iteraciones. El λ\lambdaλ que tenga el menor error promedio es el que elegiríamos, porque nos indica que el modelo tiene un buen equilibrio entre ser simple (para no sobreajustarse) y ser efectivo (para no subajustarse).

No usamos el conjunto de prueba (test) para elegir λ porque ese conjunto está reservado para medir el rendimiento final del modelo, simulando cómo funcionaría con datos completamente nuevos. Si usamos el conjunto de prueba para tomar decisiones sobre el modelo, estaríamos contaminando los resultados y perdiendo su capacidad de darnos una evaluación objetiva. La validación cruzada nos permite trabajar solo con el conjunto de entrenamiento para ajustar λ, dejando el conjunto de prueba intacto para una evaluación imparcial al final.

**3. Elección de K y Validación cruzada.**

En validación cruzada, el valor de k, es decir, el número de particiones, tiene un impacto importante en el modelo:

- **Si k es muy pequeño:** Cada fold contiene muchas muestras, pero la validación se realiza con pocos datos, lo que puede generar una estimación menos precisa del desempeño del modelo. Además, el riesgo de sobreajustar aumenta porque las particiones son menos variadas.
- **Si k es muy grande:** Cada fold contiene solo una muestra para validar, lo que da lugar a estimaciones más precisas, pero a un costo computacional muy alto porque el

modelo se entrena tantas veces como muestras hay. Además, esto puede aumentar la varianza, ya que cada validación depende de una única observación.

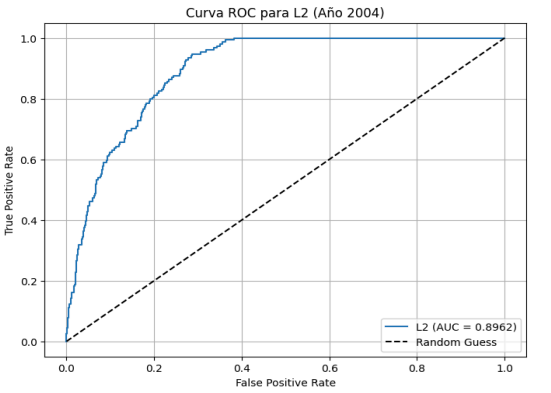
4. **Penalidades y comparación con TP3.**

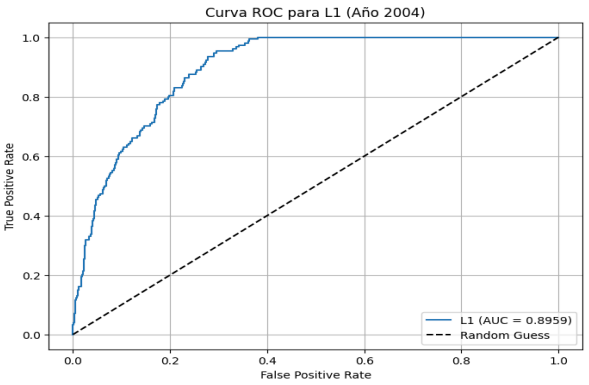
A partir de este trabajo, en el año 2004, tanto con penalización L1 como L2 presentan un accuracy idéntico del 93.38% y valores de AUC muy similares (L1=0.8959 y L2=0.8962). Las matrices de confusión revelan que la clase 0, con un soporte de 2112 instancias, se identifica correctamente con alta precisión y recall (precisión = 0.94, recall = 1.00). Esto indica que el modelo maneja con éxito la clase mayoritaria. La curva ROC está alejada de la línea de aleatoriedad lo cual es bueno, donde los VP se incrementan rápidamente con una baja tasa de FP.

Sin embargo, la clase 1, correspondiente a la minoría con solo 154 instancias, muestra un desempeño significativamente más bajo, con recall de 0.08 y un F1-score reducido (0.14 para L1 y 0.15 para L2). Esto se puede dar por un sesgo hacia la clase 0. La curva ROC muestra una menor separación, una menor capacidad predictiva.

```
Evaluación para el año 2004
Resultados para L1 (Año 2004):
Matriz de Confusión:
[[2104    8]
 [ 142   12]]
Accuracy: 0.9338
AUC: 0.8959
              precision    recall  f1-score   support

           0       0.94      1.00      0.97      2112
           1       0.60      0.08      0.14       154

    accuracy                           0.93      2266
   macro avg       0.77      0.54      0.55      2266
weighted avg       0.91      0.93      0.91      2266
```

```
Resultados para L2 (Año 2004):
Matriz de Confusión:
[[2103    9]
 [ 141   13]]
Accuracy: 0.9338
AUC: 0.8962
              precision    recall  f1-score   support

           0       0.94      1.00      0.97      2112
           1       0.59      0.08      0.15       154

    accuracy                           0.93      2266
   macro avg       0.76      0.54      0.56      2266
weighted avg       0.91      0.93      0.91      2266
```



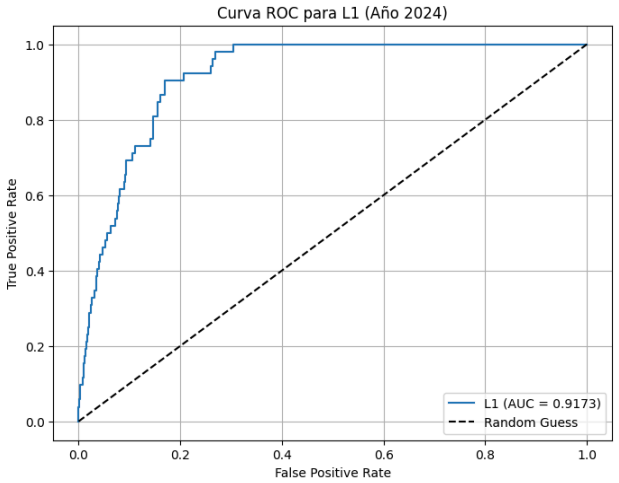

Mientras que para el 2024, L2 tiene un accuracy ligeramente superior (95.68% frente a 95.35%), pero esto se debe únicamente a su desempeño sobresaliente en la clase mayoritaria (clase 0), ignorando por completo la clase minoritaria. L1 tiene un AUC significativamente mayor (0.9173 frente a 0.7152), lo que refleja un mejor equilibrio general entre las clases. L1 tiene un desempeño mucho mejor en la clase minoritaria, identificando algunas instancias (aunque pocas), mientras que L2 no logra identificar ninguna. La curva ROC

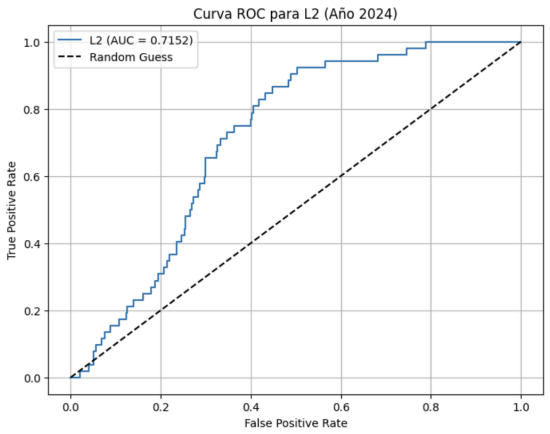
En el TP3, la regresión logística demostró que el modelo tenía una mejor capacidad para discriminar entre clases positivas y negativas, reflejado en un AUC cercano a 1. Por lo tanto,

en este caso, la performance con regularización no es mejor, ya que no logra mejorar el rendimiento en la clasificación. Sin embargo, dado que se identificaron modificaciones necesarias en el código, se concluyó que los resultados no eran completamente representativos ni confiables para una evaluación definitiva.

```
Evaluación para el año 2024
Resultados para L1 (Año 2024):
Matriz de Confusión:
[[1143    9]
 [  47    5]]
Accuracy: 0.9535
AUC: 0.9173
              precision    recall  f1-score   support

           0       0.96      0.99      0.98      1152
           1       0.36      0.10      0.15        52

    accuracy                           0.95      1204
   macro avg       0.66      0.54      0.56      1204
weighted avg       0.93      0.95      0.94      1204
```

```
Resultados para L2 (Año 2024):
Matriz de Confusión:
[[1152    0]
 [  52    0]]
Accuracy: 0.9568
AUC: 0.7152
              precision    recall  f1-score   support

           0       0.96      1.00      0.98      1152
           1       0.00      0.00      0.00        52

    accuracy                           0.96      1204
   macro avg       0.48      0.50      0.49      1204
weighted avg       0.92      0.96      0.94      1204
```




**5 y 6. Selección de λ mediante barrido y validación cruzada**

Para seleccionar el valor óptimo de λ en los modelos de regresión logística con Ridge (L2) y Lasso (L1), realizamos un barrido sobre un rango de valores definidos como λ=10^n donde n∈{−5,−4,−3,...,+5}. Este análisis se realizó utilizando validación cruzada de 10 particiones, lo que permite evaluar el desempeño del modelo en distintas particiones del conjunto de entrenamiento.

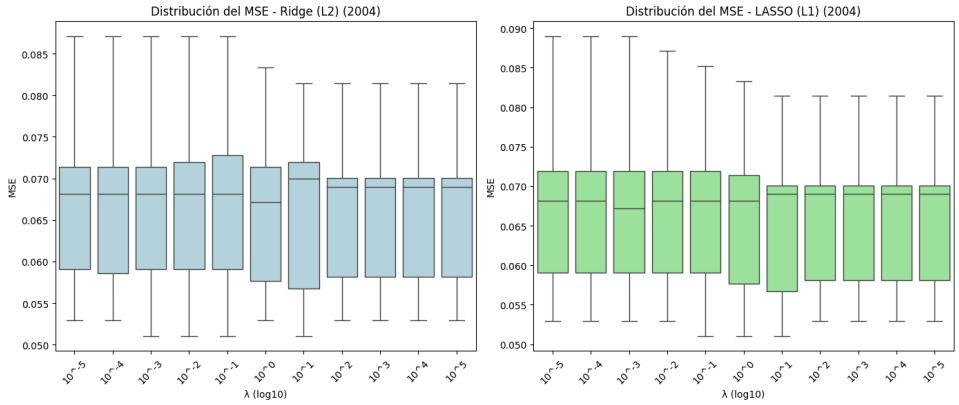
Para cada valor de λ, entrenamos los modelos Ridge y LASSO y calculamos el error medio cuadrático (MSE) promedio para cada partición de validación. En el caso de LASSO, también calculamos la proporción promedio de coeficientes eliminados (coeficientes iguales a cero) como indicador de la selección de variables realizada por el modelo.

**Resultados de λ óptimo:**

**Año 2004:** λ óptimo=10^−2 para Ridge. λ óptimo=10^−3 para LASSO.

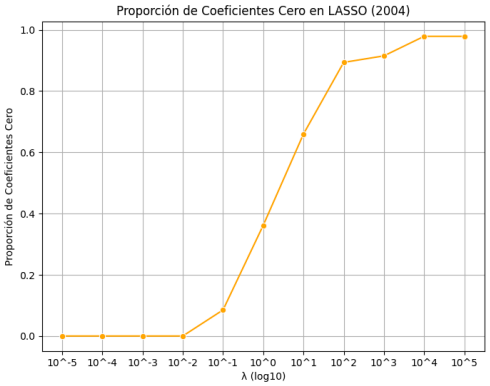
**Año 2024:** λ óptimo=10^−1 para Ridge. λ óptimo=10^-2 para LASSO.

Los gráficos de box plot muestran cómo varía el error medio de validación (MSE) en función de los valores de λ para los modelos Ridge (L2) y LASSO (L1), separados por año.

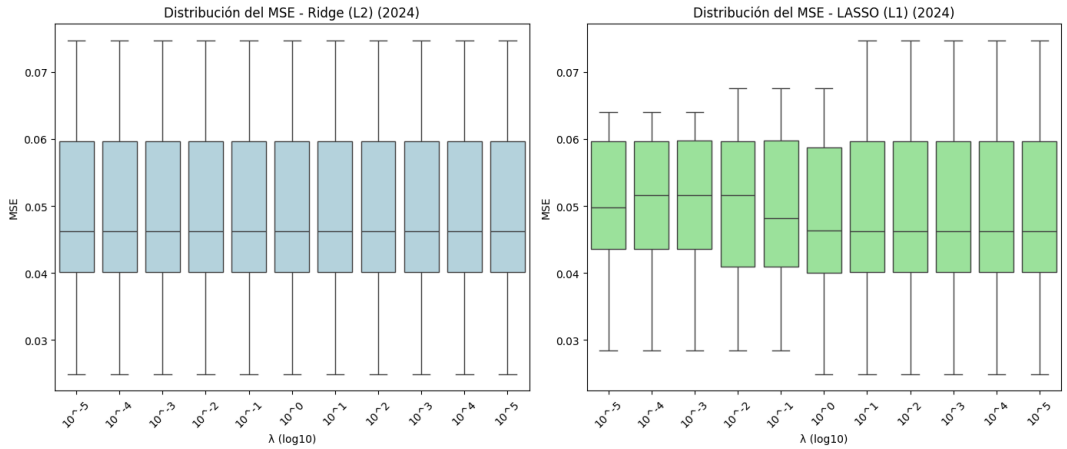



Variables descartadas por LASSO con λ óptimo (1) en 2004: ['cobertura_hogar', 'ch08_9.0', 'ch08_12.0', 'ch08_23.0', 'nivel_ed_7.0', 'cat_inac_2.0', 'cat_inac_4.0', 'cat_inac_6.0', 'cat_inac_7.0', 'iv1_4.0', 'ii7_3.0', 'ii7_6.0', 'ii7_7.0', 'ii7_9.0', 'ii7_Ocupante en relación de dependencia', 'estabilidad_vivienda_inestable'].

La selección y descarte de variables mediante LASSO está en línea con las predicciones iniciales del inciso 1, ya que las variables identificadas como buenas predictoras permanecen en el modelo (región, mas_500, pondera, iv1, iv6, iv12_3, v1, v5), a excepción de una categoría de iv1('iv1_4.0) pero esto no indica que esta variable no sea significativa, sino que simplemente esa categoría no contribuye lo suficiente al modelo.




**Análisis para el año 2024**: λ óptimo para Ridge (L2) en 2024: 1e-05. λ óptimo para LASSO (L1) en 2024: 1

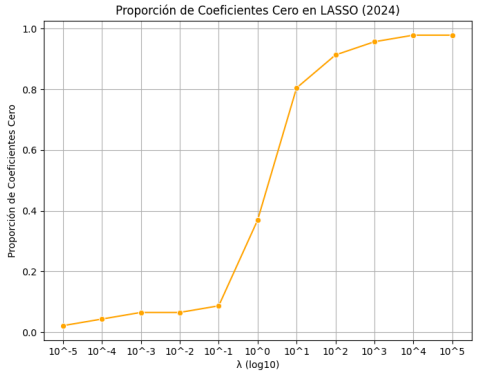



Variables descartadas por LASSO con λ óptimo (1) en 2024: ['cobertura_hogar', 'ch07_3.0', 'ch07_4.0', 'ch08_13.0', 'nivel_ed_2.0', 'nivel_ed_3.0', 'cat_inac_2.0', 'cat_inac_4.0',

'cat_inac_5.0', 'cat_inac_6.0', 'cat_inac_7.0', 'iv1_2', 'iv1_3', 'iv1_4', 'iv1_5', 'iv1_6', 'ii7_2', 'ii7_5', 'ii7_6', 'ii7_7', 'ii7_9']

En el caso del 2024, varias categorías de la variable iv, elegida en el primer punto como predictora (específicamente iv1_2, iv1_3, iv1_4, iv1_5, iv1_6) fueron descartadas, lo que indica que no todas sus categorías contribuyen de manera uniforme al modelo. Por lo que este índice no es tan relevante para la predicción de desocupación como supusimos.

En ambos casos, las variables eliminadas no forman parte de las buenas predictoras identificadas al inicio del trabajo, salvo algunas categorías de iv1.




**7. Elección de regresión logística y comparación de los años.**

En la comparación entre Ridge (L2) y LASSO (L1) en los años 2004 y 2024, ambos modelos mostraron diferencias significativas en su desempeño y en la selección de predictores relevantes. En términos de error cuadrático medio (MSE), Ridge fue más consistente en ambos años, con menor variabilidad y una mejor capacidad para mantener el desempeño incluso con cambios en λ\lambdaλ. Esto se debe a que Ridge ajusta los coeficientes sin eliminar variables, lo que asegura un modelo más estable.

Por otro lado, LASSO mostró un enfoque más agresivo al eliminar predictores menos relevantes, especialmente en 2024. En 2004, LASSO retuvo variables clave como nivel_ed (nivel educativo) e ipcf (ingreso per cápita familiar), mientras que eliminó predictores secundarios como algunas categorías de iv1 (tipo de vivienda). En 2024, fue aún más selectivo, eliminando un mayor número de variables y reteniendo únicamente las más significativas, lo que podría reflejar cambios estructurales en la dinámica socioeconómica entre ambos años.

Aunque Ridge presentó un desempeño más estable, LASSO resultó más efectivo para simplificar el modelo y reducir su complejidad, especialmente en 2024, cuando la regularización ayudó a ajustar mejor el modelo a datos más complejos. Esto sugiere que, dependiendo del objetivo del análisis, Ridge es preferible si se busca estabilidad, mientras que LASSO es más adecuado para obtener interpretabilidad al seleccionar únicamente los predictores más relevantes.