# **Trabajo práctico 1: Aprendizaje Automático**

## ***Modelos Clásicos***

Mastelli - Mujica - Pengue- Tiraboschi

# **Introducción:**

La capacidad de predecir el precio de una propiedad tiene un impacto significativo en las decisiones financieras, la inversión, la planificación urbana y el desarrollo de políticas. Proporciona una base sólida para tomar decisiones informadas y maximizar el valor de los activos inmobiliarios. En este trabajo se analizará la información de los inmuebles de Capital Federal que se encuentran en venta, a modo de generar modelos de Machine Learning para predecir el precio de propiedades desconocidas. Para esto se procederá a utilizar tres modelos de regresión para predecir la variable price y dos modelos de clasificación para predecir la variable llamada price\_respuesta.

La capacidad de predecir el precio de una propiedad ha sido objeto de numerosos estudios y análisis en la literatura especializada ("La valoración de propiedades es una herramienta valiosa para los compradores, vendedores e inversores inmobiliarios." - The Appraisal Journal). Sin embargo, a pesar de estos avances, existe un vacío en la investigación con respecto a cómo los modelos de Machine Learning pueden utilizarse para predecir el precio de propiedades desconocidas en un contexto específico, como el de los inmuebles de Capital Federal en venta. En este trabajo, nos proponemos llenar ese vacío y demostrar cómo los modelos de Machine Learning pueden generar predicciones precisas del precio de las propiedades en base a la información disponible, proporcionando así una valiosa herramienta para la toma de decisiones en el mercado inmobiliario.

Una limitación de los árboles de decisión es que la división del espacio de entrada se basa en divisiones duras en las que solo un modelo es responsable de hacer predicciones para cualquier valor dado de las variables de entrada (1). Además, las divisiones en un árbol de decisión son difíciles, por lo que cada región del espacio de entrada está asociada con un modelo de nodo hoja, y solo uno. El último problema es particularmente problemático en la regresión, donde normalmente buscamos modelar funciones suaves y, sin embargo, el modelo de árbol produce predicciones constantes por partes con discontinuidades en los límites de división (1). El bosque aleatorio supera el problema de la alta variación en un árbol individual al proporcionar los dos tipos de aleatoriedad mencionados anteriormente. Las muestras de subconjuntos aleatorios cometen diferentes errores y, por lo tanto, los estimadores generalizan bien al tomar el promedio uniforme de cada predictor que ayuda a cancelar los errores (2).

En base a estos antecedentes la hipótesis planteada para los modelos de regresión es que el uso del modelo de Random Forest para regresión logrará una mayor precisión en la predicción de valores numéricos continuos en comparación con arboles de decision y regresion lineal. Esto se basa en que utiliza múltiples árboles de decisión y combina sus predicciones, lo que le permite ajustarse mejor a los datos de entrenamiento y generalizar de manera más precisa a nuevos datos. Una de las ventajas de Random Forest es que reduce la varianza en comparación con un solo árbol de decisión. Esto se debe a que cada árbol se entrena con una muestra aleatoria y utiliza características aleatorias para tomar decisiones en cada división del árbol. Al combinar los resultados de múltiples árboles, el efecto de las decisiones incorrectas o sesgadas tomadas por un solo árbol se amortigua, lo que conduce a una reducción en la varianza general. El control de la varianza en Random Forest se logra mediante el promedio de las predicciones de múltiples árboles. En lugar de depender de un solo árbol, el modelo toma una decisión basada en la agregación de las predicciones de todos los árboles del bosque. Esto ayuda a mitigar el sobreajuste y a mejorar la generalización del modelo, ya que las decisiones individuales de cada árbol se promedian. Además, Random Forest también permite controlar la varianza mediante la selección aleatoria de características en cada división de un árbol. En lugar de utilizar todas las características disponibles, se toma un subconjunto aleatorio, lo que reduce la correlación entre los árboles y disminuye la varianza general del modelo. Con respecto a los modelos de clasificación, la hipótesis que se plantea es que el uso del modelo Random Forest superará en precisión al Árbol de decisión. El Random Forest utiliza múltiples árboles de decisión y combina sus predicciones, lo que le permite capturar relaciones más complejas entre las características y las etiquetas de clase.

# **Materiales y métodos:**

Para estudiar el problema se trabajó con un dataset preprocesado con datos sobre propiedades de Capital Federal provenientes de la página web de venta de propiedades Properati.

Los datos fueron analizados con el objetivo de predecir el precio de las viviendas.

La base de datos contaba de 142300 propiedades sin datos faltantes. Las mismas estaban caracterizadas de acuerdo al número de cuartos (rooms), baños (bathrooms), superficie total (surface\_total), precio (price),tipo de propiedad (property\_type), presencia de terraza. patio o balcón (terraza\_patio\_balcon), barrio (l3), comunas, price\_respuesta que fueron usadas como variables predictoras.

Tres son numéricas (rooms, bathrooms, surface\_total) y cuatro son categóricas (property\_type, terraza\_patio\_balcon, l3, comunas).

De las variables numéricas, rooms y bathroom (cantidad de ambientes y cantidad de baños de la vivienda) toman ambas valores de 1 a 5. El rango de surface\_total (superficie total de la propiedad) va desde 11 a 8900 metros cuadrados.

Se contabilizan 6 tipos distintos de vivienda en el atributo property\_type: Departamento (81%), PH (9%), Casa(4%), Local comercial (3%), Oficina (3%) y Casa de campo. Terraza\_patio\_balcon es una variable categórica dicotómica booleana que indica si el inmueble posee una de esas características (si tiene o no terraza, patio y/o balcón), aproximadamente la mitad de las propiedades ofrecidas a la venta cuentan con alguna de esas características. El atributo l3 corresponde al barrio en donde está ubicada la vivienda (57 valores distintos) y el atributo comuna refiere a la unidad administrativa que corresponde a la variable l3, las comunas abarcan cada una a más de un barrio (valores de 1 a 15).

Entre los barrios, los seis que contabilizan más propiedades son los siguientes: Palermo (13%), Belgrano (11,4%), Caballito (6,8%), Recoleta (5,4%), Almagro y Villa Urquiza (5%). En tanto que las comunas 13, 14, 12, 2 y 1 acumulan el 55% de las ofertas.

Las dos columnas restantes son las variables target : la variable price, numérica (de 15.500 a 17.600.000), se utilizará con los algoritmos de regresión y la otra, price\_respuesta categórica tricotómica (bajo, medio, alto), se usará para los de clasificación. Para la variable price\_respuesta el porcentaje relativo de cada etiqueta es de 36%, 35% y 29% para bajo, medio y alto por lo que podemos decir que los datos están balanceados para esa categoría.

Los valores atípicos y las distribuciones de las variables predictoras se identifican y grafican en lenguaje python en el colab, el cual es un producto de Google que permite escribir y ejecutar código directamente desde el navegador.

Para realizar las predicciones eliminamos la variable comunas por considerarla redundante de la información que proporciona el barrio (columna l3). También eliminamos los registros duplicados.

## **Análisis estadístico / modelos**

Para generar un modelo de predicción para la variable continua “Price” decidimos utilizar modelos de regresión, cuyo objetivo principal es justamente predecir valores continuos en base a la información proporcionada por una o varias variables independientes. Por otra parte, para el problema de clasificación de predicción de la variable “price\_respuesta” decidimos utilizar un modelo de árbol de decisión.

Se tomó la decisión de ejecutar y comparar un modelo de regresión lineal múltiple clásico (aplicando dos variantes de regularización) y el modelo simple del árbol de decisión con un modelo ensamble para cada problema (en este caso, Random Forest3 tanto para regresión como para clasificación). Esto se hizo con la hipótesis de que los modelos de ensamble, permitirán obtener mejores resultados en cuanto a Raíz del Error Cuadrático Medio (RECM) para la regresión y accuracy (clasificación) que los modelos más simples.

**Regresión Lineal Múltiple**.

Como los datos presentaban ciertos valores considerados outliers, decidimos utilizar los modelos de regresión Ridge y Lasso para poder llevar a cabo una regularización (es decir, penalizar estos valores atípicos para que no tengan tanto peso en el modelo final).

Al aplicar estos modelos, decidimos realizar una búsqueda para encontrar el hiperparámetro Alpha óptimo utilizando validación cruzada. Alpha, también llamado término de penalización, está diseñado para penalizar la complejidad del modelo: cuanto mayor es el Alpha, menos complejo será el modelo obtenido, disminuyendo el error debido a la varianza o sobreajuste. Sin embargo, si el Alpha es demasiado grande podemos estar llevando a un subajuste e introduciendo un mayor sesgo.

Creamos un rango de valores para Alpha en una escala logarítmica. Luego, utilizamos el objeto RidgeCV para implementar la regresión Ridge con validación cruzada de 5 divisiones, pasando el rango de valores de Alpha. Del mismo modo, se utiliza el objeto Lasso para implementar la regresión Lasso con validación cruzada de 5 divisiones, también pasando el rango de valores de Alpha. Ambos modelos se ajustan a los datos de entrenamiento utilizando la función fit. Al finalizar, se obtendrán los modelos Ridge y Lasso ajustados con el valor óptimo de Alpha encontrado mediante la validación cruzada.

**Árbol de decisión para clasificación.**

Los árboles de decisión son un modelo de aprendizaje no paramétrico que puede utilizarse tanto para tareas de regresión como de clasificación. Como su nombre lo indica, tiene una estructura jerárquica compuesta de un nodo raíz, ramas, nodos internos y hojas. Cada nodo interno representa una decisión basada en una característica específica, y cada hoja (nodo terminal) representa el resultado o predicción.

Para este modelo se probó una variedad de combinaciones de hiperparámetros:

* Criterio de división(criterion): determina la métrica utilizada para evaluar la calidad de una división. Los dos criterios que probamos son "gini", que utiliza el índice Gini como medida de impureza y "entropy", que utiliza la entropía como medida de impureza.
* Profundidad máxima del árbol (max\_depth): establece la profundidad máxima que puede tener el árbol. Limitar la profundidad ayuda a evitar el sobreajuste y mejora la generalización del modelo.
* Número mínimo de muestras por hoja (min\_samples\_leaf): representa el número mínimo de muestras requeridas para formar una hoja en el árbol.

Utilizamos validación cruzada estratificada para evaluar el rendimiento del modelo en diferentes divisiones de los datos y ajustamos el modelo a los datos de entrenamiento, imprimiendo el diccionario de los mejores hiperparámetros encontrados y la precisión del modelo correspondiente. El objetivo es encontrar los hiperparámetros que maximicen la accuracy del modelo.

**Random Forest**.

Random Forest se basa en la idea de crear un conjunto o “bosque” de árboles de decisión y combinar sus resultados para obtener predicciones más precisas y robustas. Para tareas de clasificación, la predicción final de Random Forest se determina por votación de los árboles. Cada árbol emite su propia predicción y la clase más frecuente entre todos los árboles se selecciona como la predicción final. Para tareas de regresión, se toma el promedio de las predicciones de todos los árboles.

Los hiperparámetros optimizados fueron los mismos descritos para el árbol de decisión en el apartado anterior.

**Resultados:**

Al correr los diferentes modelos de regresión y clasificación a modo tal de concluir sobre las hipótesis planteadas se obtuvieron los siguientes resultados:

Respecto de los modelos de regresión, el peor rendimiento se observó para el Árbol de decisión. En los modelos de regresión líneal, ambas regularizaciones arrojaron resultados similares ya que el Alfa óptimo obtenido por CV fue un valor cercano al 0 por lo que el peso de la regularización es despreciable. Finalmente, el modelo de Random Forest arrojó valores de desempeño en el conjunto de testeo mayores que el observado para los modelos clásicos (ver Tabla 1).

Respecto de los modelos de clasificación, al igual que para regresión, se observa que el modelo Random Forest tuvo una mejor performance que el modelo Árbol de decisión (ver Tabla 2).

***Tabla 1 Resultados de los modelos de regresión***

| Tipo de Modelo | Train - RECM | Test - RECM |
| --- | --- | --- |
| Árbol de decisión | 220.200 | 211.107 |
| Regresión lineal con regularización Ridge | 176.346 | 184.185 |
| Regresión lineal con regularización Lasso | 176.346 | 184.185 |
| Random Forest | 176.845 | 164.488 |

***Tabla 2 Resultados de los modelos de clasificación***

| Tipo de Modelo | Train - Accuracy | Test - Accuracy |
| --- | --- | --- |
| Árbol de decisión | 0,7441 | 0,7458 |
| Random Forest | 0,7528 | 0,7515 |

# **Discusión:**

Los resultados obtenidos al evaluar los distintos algoritmos de regresión mostraron diferencias en sus respectivos desempeños. La RECM se utilizó como métrica para medir la precisión de los modelos en el conjunto de testeo.

Comparando los resultados se observó que los modelos de regresión lineal (Ridge y Lasso) lograron un desempeño similar en términos de RECM al Random Forest, el que a priori era el que iba a presentar la mejor performance. Estos hallazgos coinciden con estudios anteriores (Hastie et al. 2009) que sugieren que los modelos de regresión lineal pueden ser efectivos en la resolución de problemas de regresión.

Sin embargo, es importante destacar que el Árbol de decisión obtuvo un RECM más alto que los otros modelos evaluados. Esto indica que este algoritmo tuvo dificultades para capturar la relación subyacente en los datos y, por lo tanto, presentó un desempeño inferior. Estos resultados son consistentes con la literatura (Hastie et al. 2009), que señala que los árboles de decisión pueden tener limitaciones en la resolución de problemas de regresión cuando se enfrentan a conjuntos de datos complejos o con relaciones no lineales.

En su libro "The Elements of Statistical Learning" Hastie plantea que “...los árboles de decisión son una herramienta poderosa para la regresión, pero pueden tener algunas limitaciones. Pueden ser propensos al sobreajuste, especialmente si el conjunto de datos es pequeño o tiene muchas características. Además, pueden tener dificultades para capturar relaciones no lineales entre los predictores y la respuesta. Para abordar estas limitaciones, a menudo es útil usar un conjunto de árboles pequeños, en lugar de un árbol grande, y usar una técnica llamada poda para eliminar ramas del árbol que no son importantes para la predicción. También se pueden usar otras técnicas de aprendizaje automático, como los bosques aleatorios, que pueden ser más resistentes al sobreajuste que los árboles de decisión individuales".

En cuanto a los algoritmos de clasificación, se evaluaron el Árbol de decisión y el Random Forest utilizando el accuracy como métrica de evaluación.

Los resultados mostraron que de ambos modelos de clasificación el Random Forest tuvo una mejor precisión que el Árbol de decisión. Aunque el accuracy no es la única métrica para evaluar el desempeño de los modelos de clasificación, los resultados sugieren que ambos modelos pueden ser adecuados para el problema de clasificación teniendo mayor precisión el Random Forest.

# **Conclusión:**

Basados en los resultados obtenidos, se puede concluir sobre las hipótesis planteadas para ambos tipos de modelos (regresión y clasificación). Se observó que la Regresión Lineal con regularización Lasso o Ridge logró similar precisión en la predicción de los valores numéricos continuos de entrenamiento en comparación al Random Forest. Y que el peor performante fue el Árbol de decisión.

La menor puntuación en el RECM en testeo indica que el modelo de Random Forest ajustó mejor a los datos nuevos y tuvo una capacidad más precisa de generalizar a los mismos. Por lo tanto, se puede aceptar la hipótesis inicial planteada.

Estos resultados resaltan la eficacia de Random Forest en la resolución de problemas de regresión, particularmente cuando se requiere una alta precisión en la predicción de valores continuos.

En términos de métodos de clasificación, se pudo concluir que el Random Forest superó al Árbol de decisión en precisión. Esto implica que el Random Forest logró realizar predicciones más precisas que el Árbol de decisión en el conjunto de datos evaluado, demostrando así su capacidad para clasificar con mayor precisión las etiquetas de clase discretas en comparación con el Árbol de decisión. Por lo tanto, los resultados obtenidos respaldan nuevamente la hipótesis inicial planteada, en este caso, para clasificación.

# Bibliografía:

1 - Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer, Capítulo 14.4: Decision Trees for Regression.

2 - Shashi Bhushan Jha, Radu F. Babiceanu (2020). Housing Market Prediction Problem using Different Machine Learning Algorithms: A Case Study, 4.2. Random Forest Method.

3- Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324