Universidad del Valle de Guatemala
Facultad de Ingeniería

Data Science
Departamento de Ciencias de la Computación
Ciclo II, 2024

Laboratorio 8

José Ricardo Méndez González, 21289
Sara María Pérez Echeverría, 21371

Guatemala, 20 de octubre de 2024

Introducción

El trabajo realizado se centró en el procesamiento de un conjunto de datos sobre viviendas en Brasil, con el objetivo de desarrollar un modelo predictivo que permita estimar el precio de una casa con base en diferentes características. Durante el proyecto, se entrenaron y evaluaron tres modelos diferentes: Regresión Lineal, Random Forest y Gradient Boosting. Posteriormente, se desarrolló una aplicación web para interactuar con el modelo de predicción final utilizando un formulario HTML en el frontend y un backend basado en Flask.

Procesamiento de Datos

El primer paso en el proyecto fue la limpieza y preprocesamiento de los datos. Las variables más relevantes, como el tamaño de la vivienda, la cantidad de habitaciones, la ubicación, y otros factores, se seleccionaron para incluirlas en el conjunto de entrenamiento. Entre las tareas de preprocesamiento realizadas se incluyeron:

1. **Limpieza de datos faltantes:** Se eliminó o imputaron valores nulos para evitar sesgos en el modelo.
2. **Normalización y escalado de características:** Esto fue crucial para asegurar que los algoritmos basados en gradientes, como la Regresión Lineal y Gradient Boosting, se entrenaran de manera eficiente.
3. **Codificación de variables categóricas:** Características como la ubicación se transformaron en variables numéricas usando técnicas como One-Hot Encoding.

Este paso fue fundamental para asegurar la calidad de los datos y preparar un conjunto robusto para el entrenamiento de los modelos.

Modelos Entrenados

Tres modelos distintos se entrenaron con el objetivo de encontrar el mejor ajuste y precisión en la predicción de los precios de las casas:

1. **Regresión Lineal:** Este modelo fue el punto de partida para el proyecto, ya que es un modelo estadístico clásico que sirve como referencia. La Regresión Lineal demostró ser simple y eficiente, pero su rendimiento fue limitado debido a la naturaleza no lineal de las relaciones entre las características y el precio de las viviendas.
2. **Random Forest:** Este modelo basado en árboles de decisión fue una mejora significativa en comparación con la Regresión Lineal. Random Forest es robusto y captura relaciones no lineales entre las características. Sin embargo, el modelo tiende a ser menos interpretativo y más costoso en términos computacionales, lo que se notó durante el entrenamiento.
3. **Gradient Boosting:** Finalmente, se implementó Gradient Boosting, que utiliza un enfoque basado en el error residual para mejorar la precisión del modelo. Este modelo ofreció el mejor desempeño en términos de error cuadrático medio (MSE) y precisión general, superando tanto a la Regresión Lineal como a Random Forest. Este fue el modelo elegido debido a su capacidad para manejar relaciones complejas y su equilibrio entre precisión y tiempo de entrenamiento.

Evaluación y Selección del Modelo

Después de entrenar los tres modelos, se realizó una evaluación comparativa utilizando métricas de rendimiento como el MSE y el coeficiente de determinación (R^2). Gradient Boosting mostró los mejores resultados, con el menor error en la predicción de precios y una alta capacidad para generalizar sobre datos de prueba.

El proceso de selección del modelo incluyó la validación cruzada para evitar el sobreajuste, asegurando que el modelo elegido fuera capaz de realizar predicciones precisas sobre datos no vistos.

Implementación del Formulario Web

Una vez seleccionado el modelo de Gradient Boosting, se diseñó una aplicación web para permitir que los usuarios interactuaran con el modelo de manera sencilla. La aplicación se dividió en dos componentes principales:

1. **Frontend (HTML y CSS):** Se desarrolló un formulario web sencillo donde el usuario puede ingresar características de una casa, como el tamaño, el número de habitaciones, la ubicación, entre otros. Este formulario fue diseñado para ser intuitivo y fácil de usar, con validaciones básicas en el frontend.
2. **Backend (Flask):** Flask se utilizó para desarrollar la lógica del backend. El formulario HTML envía los datos ingresados por el usuario al servidor Flask, donde el modelo de Gradient Boosting cargado realiza la predicción del precio de la vivienda en función de los inputs. Flask fue elegido debido a su simplicidad y capacidad para manejar fácilmente peticiones HTTP, lo que permitió un rápido desarrollo e integración del modelo de predicción.

Desafíos Encontrados

Durante el proyecto, surgieron varios desafíos que afectaron tanto al desarrollo del modelo como a la implementación web:

- **Preprocesamiento de datos:** El manejo de valores nulos y variables categóricas fue un desafío, ya que la información geográfica (ubicación) tenía una alta correlación con los precios de las viviendas. Hubo que encontrar el balance adecuado para evitar el sobreajuste del modelo en estas características.
- **Despliegue del modelo:** Integrar el modelo con Flask y asegurar que el formulario web interactuara correctamente con el backend fue un proceso que implicó varias iteraciones. Se aseguraron medidas de seguridad para la entrada de datos del usuario y para garantizar una comunicación fluida entre el frontend y el backend.

Conclusión

Este proyecto proporcionó una valiosa experiencia en el desarrollo de un sistema completo de predicción de precios de casas, desde el procesamiento de datos y entrenamiento de modelos hasta la implementación de una aplicación web. El modelo de Gradient Boosting demostró ser la mejor opción, superando a la Regresión Lineal y a Random Forest en precisión.

Además, el desarrollo de una aplicación web con Flask permitió hacer accesible el modelo a través de una interfaz amigable para el usuario, brindando una solución práctica y funcional para la predicción de precios. Este proyecto destaca la importancia de la elección adecuada del modelo y la integración eficiente entre el frontend y el backend en el desarrollo de aplicaciones de machine learning.