E-commerce Furniture Sales Prediction Project

This project was completed during my internship, where I worked on predicting how well furniture items sell online. The main idea was to clean the data, understand what drives sales, and build a machine learning model that can predict how many units of a furniture item might sell based on details like its price, original price, and shipping type.

Project Objective

The goal was to create a simple and clear machine learning project where we:

- Understood the e-commerce furniture dataset
- Cleaned and prepared the data
- Explored it to find patterns
- Built a model to predict the number of items sold

About the Dataset

We used a dataset named **ecommerce_furniture_dataset_2024.csv**, which includes details about various furniture products sold online. The key columns were:

- productTitle: Name of the product
- originalPrice: The original price before discount
- price: The final selling price
- sold: Number of units sold (this is going to be our prediction target for ML mode)
- tagText: Extra product info like whether shipping was free

Step-by-Step Project Process

1. Importing and Viewing the Data

We started by loading the dataset and checking how many rows and columns it had. This gave us an idea of what we were working with. We also checked the basic structure, such as data types and column names.

While doing that, we noticed that the price-related columns had special characters like dollar signs and commas, so we cleaned those to turn them into proper numbers.

---

2. Data Cleaning and Preparation

To make the data ready for machine learning, we cleaned it step by step:

- **Removed commas and symbols** from price and originalPrice, and converted them to float type
- **Handled missing values**:
    - Filled missing values in originalPrice with the median, because there were outliers and we didn't want to remove a lot of rows
    - Replaced missing tagText values with the word "unknown"
- **Removed duplicate rows** to avoid errors during modeling
- After cleaning, the dataset was ready for analysis

---

3. Feature Engineering

I created two new columns in order to improve the model:

- total_revenue: Calculated as price * sold to know how much money each product made
- discounted_%: Calculated how much discount each product got. This helped us understand whether discounting had any effect on sales. We made sure discount values stayed between 0 and 100

We also encoded the tagText column using **Label Encoding**, so machine learning models could use it.

For productTitle, we used **TF-IDF (Term Frequency-Inverse Document Frequency)**. This technique helped convert product names into numeric values while keeping the meaning of the words. Instead of just giving a number to each title, it focused on which words were important, like "ergonomic" or "folding."

---

4. Splitting the Data

We split the data into two parts:

- **Training set** (used to train the model)
- **Testing set** (used to check how well the model works)

This helped us make sure our model wasn't just memorizing the data, but actually learning from it.

---

5. Machine Learning Models

We used two models to predict the number of units sold:

a. Random Forest Regressor

This model works by combining multiple decision trees. It is very good at handling both numerical and categorical features, and it works well even when there are outliers. It gave us a strong baseline model.

b. GridSearchCV with Random Forest

After building the first model, we used GridSearchCV to improve it. This tool automatically tested different settings (like how many trees to use) and picked the best combination. This helped improve the model's accuracy and reduce overfitting.

---

6. Evaluating the Model

We checked the model's performance using:

- **MSE (Mean Squared Error)**: Tells how far off our predictions were
- **R^2 Score**: Shows how well the model explained the data

The Random Forest with GridSearchCV gave us better results than simple linear regression, which showed that it was a good fit for this kind of problem.

Final Thoughts

This project was a good way to understand how to build a machine learning model from scratch using real e-commerce data. We cleaned and explored the data, created new features, converted text into numbers, and built models that could actually make predictions.

This kind of work is useful for any business that wants to understand its sales patterns and make better decisions based on data.