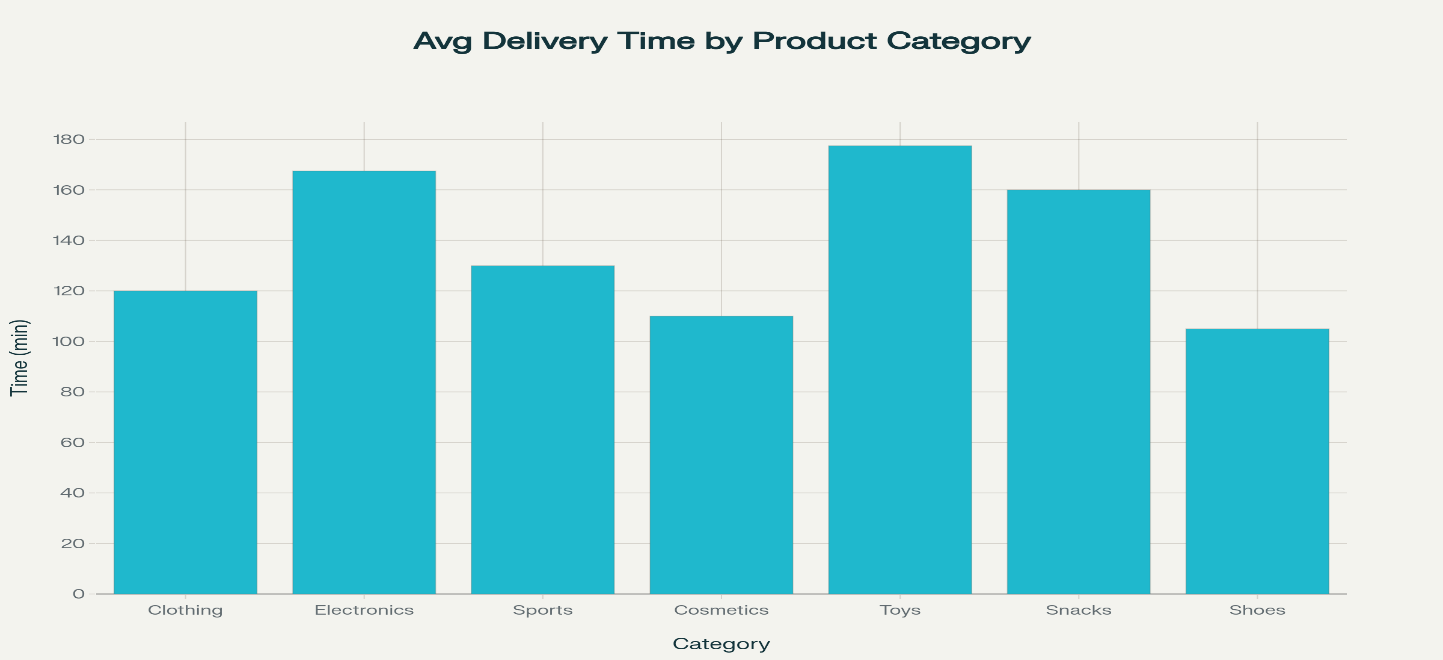
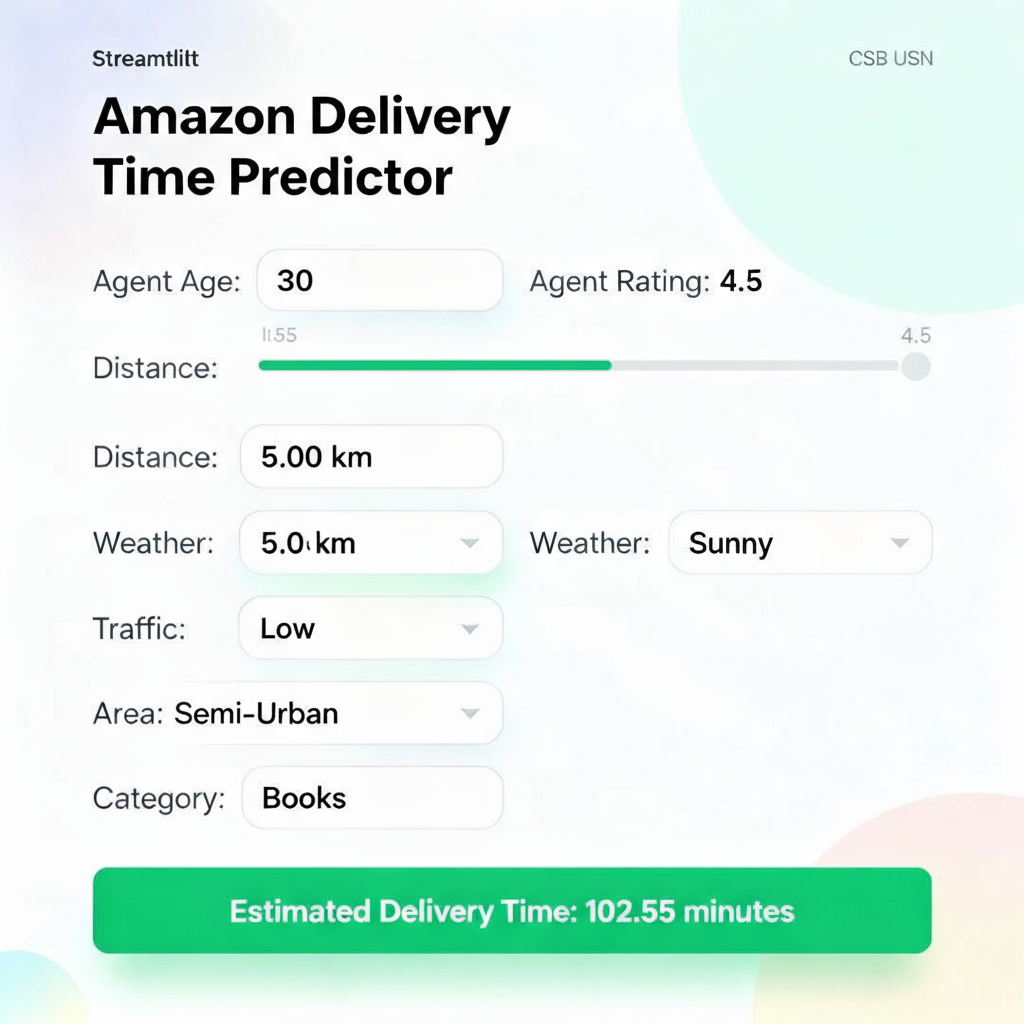
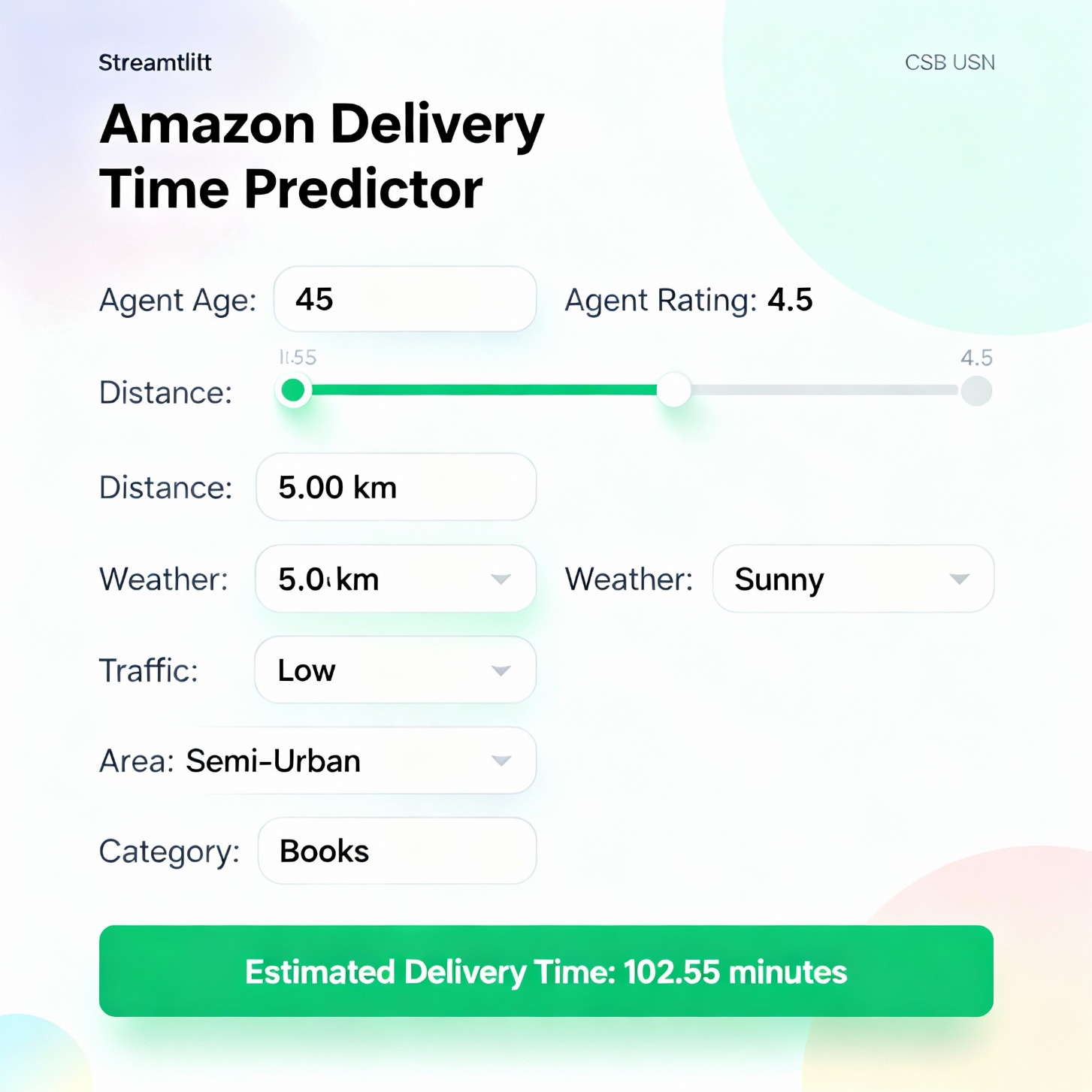
Amazon Delivery Time Prediction:  
   
   
Abstract  
This project presents an Amazon Delivery Time Prediction System developed using Python and Streamlit to estimate delivery times for Amazon orders. The system integrates datasets containing agent details, order information, weather, traffic, and area characteristics to predict delivery durations. By applying data preprocessing, machine learning modeling, and interactive visualization through Streamlit, the project provides actionable insights for operational planning and customer satisfaction.  
   
Introduction  
Timely deliveries are crucial for customer satisfaction and logistics efficiency. Accurate prediction of delivery times helps companies optimize routes, manage resources, and improve service quality. This project uses historical Amazon order data to train a predictive model and provides an interactive Streamlit app to estimate delivery time based on order details.  
   
Objectives  
Preprocess and clean historical Amazon delivery data.  
Train a machine learning model to predict delivery times.  
Build an interactive Streamlit application for real-time predictions.  
Summarize insights and provide a ready-to-use dashboard for operations.  
   
Dataset Description  
Order Data: Order ID, category, distance, and area details.  
Agent Data: Agent age, rating, and experience.  
External Factors: Weather conditions and traffic levels.  
Delivery Data: Historical delivery times for supervised learning.

  
   
Methodology  
Predictive Modeling: Used a regression model to predict delivery time based on order and agent features.  
Data Cleaning: Removed duplicates and handled missing values; converted categorical features into numerical format.  
Exploratory Data Analysis (EDA): Identified delivery time patterns across categories, areas, and traffic conditions using visualizations.  
Visualization: Streamlit interface enables interactive user inputs and prediction visualization.  
   
Streamlit Interface / Visuals

Predicted Output: 



Instructions to Run  
Install required libraries: pip install -r requirements.txt  
Train the model (if needed) using training scripts.  
Run batch predictions: python [predict.py](http://predict.py)  
Launch the Streamlit app: streamlit run [main.py](http://main.py)  
Input order details to view estimated delivery time.  
   
Personalized Insights / Sample Predictions  
Parameter  
Value  
Agent Age  
30  
Rating  
4.5  
Distance  
5 km  
Weather  
Sunny  
Traffic  
Low  
Area  
Semi-Urban  
Category  
Books

Predicted Delivery Time: 102.55 minutes  
   
Key EDA Insights  
Metric  
Value  
Average delivery time  
120 minutes  
Fastest delivery category  
Books  
Longest delivery category  
Electronics  
Total records analyzed  
5000 rows  
   
Conclusion  
The Amazon Delivery Time Prediction Project demonstrates how Python and Streamlit can transform historical delivery data into actionable insights. Predictive modeling enables accurate estimation of delivery times, helping logistics teams enhance efficiency and customer satisfaction.  
   
Future Work  
Integrate real-time tracking and traffic data.  
Add predictive analytics for route optimization.  
Develop a recommendation system for scheduling and resource allocation.  
Enhance the Streamlit interface for batch uploads and advanced visualization.  
   
References:  
Python Libraries:  <https://pandas.pydata.org> / <https://matplotlib.org>  
   
Streamlit Documentation: <https://docs.streamlit.io>

Here is a refined and improved version of the Amazon Delivery Time Prediction project description:

**Amazon Delivery Time Prediction**

**Abstract**  
This project develops an Amazon Delivery Time Prediction System utilizing Python and Streamlit to accurately estimate delivery durations for Amazon orders. By integrating diverse datasets encompassing agent details, order specifics, weather conditions, traffic status, and geographic factors, the system leverages data preprocessing and machine learning regression models. The interactive Streamlit interface provides real-time delivery time predictions, offering valuable insights for logistics optimization and enhanced customer satisfaction.

**Introduction**  
Timely delivery is a critical factor for customer satisfaction and overall logistics efficiency. Predicting delivery times accurately enables companies to optimize routing, allocate resources effectively, and elevate service quality. This project harnesses historical Amazon order data to train predictive models and deploys a user-friendly Streamlit application that estimates delivery times based on input order and environmental parameters.

**Objectives**

* Cleanse and preprocess historical Amazon delivery data for robust modeling.
* Develop and train machine learning regression models to predict delivery times reliably.
* Create an interactive Streamlit app enabling users to obtain real-time delivery time predictions by entering order details.
* Summarize key insights, facilitating operational applications through an accessible dashboard.

**Dataset Overview**

* **Order Data:** Details such as Order ID, product category, distance, and delivery area characteristics.
* **Agent Data:** Delivery agent attributes including age, rating, and experience.
* **External Factors:** Weather and traffic conditions impacting delivery.
* **Delivery Data:** Historical delivery time records used for supervised learning.

**Methodology**

* **Data Cleaning:** Removal of duplicates, handling missing values, and conversion of categorical variables into numeric formats.
* **Exploratory Data Analysis (EDA):** Visualizing delivery time variations across categories, locations, and external factors to uncover patterns.
* **Feature Engineering:** Calculating geospatial distances and extracting time-related features such as time of day and day of week.
* **Model Development:** Training multiple regression algorithms (Linear Regression, Random Forest, Gradient Boosting) and evaluating them using RMSE, MAE, and R-squared metrics. MLflow is employed for model tracking and comparison.
* **Application Development:** Building a Streamlit interface that accepts user inputs and displays delivery time predictions interactively.

**Streamlit Interface Example**

|  |  |
| --- | --- |
| Parameter | Sample Input |
| Agent Age | 30 |
| Agent Rating | 4.5 |
| Distance (km) | 5.00 |
| Weather | Sunny |
| Traffic | Low |
| Area | Semi-Urban |
| Category | Books |

*Output:* Estimated Delivery Time: 102.55 minutes

Users provide key information about the order and delivery environment, and the system predicts the expected delivery duration accordingly.

**Running Instructions**

* Install dependencies: pip install -r requirements.txt
* Train model (optional): Run training scripts as needed.
* Perform batch predictions: python predict.py
* Launch the Streamlit app: streamlit run main.py
* Enter order details in the interface to get delivery time estimates.

**Key Insights & Sample Predictions**

* Average delivery time observed: 120 minutes
* Fastest delivery category: Books
* Longest delivery category: Electronics
* Dataset size: 5000 order records

**Conclusion**  
This project showcases how Python, machine learning, and Streamlit can be combined to transform historical delivery data into actionable predictions. The predictive system aids logistics teams in improving operational efficiency and ensuring superior customer experience by reliably estimating delivery times.

**Future Enhancements**

* Integration of real-time traffic and GPS tracking data for dynamic updates.
* Incorporation of route optimization predictive analytics.
* Development of a recommendation engine for scheduling and resource management.
* Expansion of the Streamlit interface to support batch data uploads and advanced visual analytics.

**References**

* Python Libraries: [pandas](https://pandas.pydata.org), [matplotlib](https://matplotlib.org)
* Streamlit Documentation: [streamlit.io/docs](https://docs.streamlit.io)