|  |  |
| --- | --- |
| Project Title | Backorder Prediction |
| Technologies | Machine Learning |
| Technology Domain | E-commerce |
| Project Difficulties | level Intermediate |

**Project: Backorder Prediction**

**Problem Statement**: Backorders are unavoidable, but by anticipating which things will be back-ordered, planning can be streamlined at several levels, preventing unexpected strain on production, logistics, and transportation. ERP systems generate a lot of data (mainly structured) and also contain a lot of historical data; if this data can be properly utilized, a predictive model to forecast backorders and plan accordingly can be constructed. Based on past data from inventories, supply chain, and sales, classify the products as going into backorder (Yes or No).

**Approach**: The classical machine learning tasks like

* Data Exploration,
* Data Cleaning,
* Feature Engineering,
* Model Building
* Model Testing.

Try out different machine learning algorithms that are best fit for the above case.

**Results**: You have to build a solution that should be able to predict the backorder sales for a particular product according to the provided dataset.

**Project Evaluation metrics:**

Code:

• You are supposed to write a code in a modular fashion

• Safe: It can be used without causing harm.

• Testable: It can be tested at the code level.

• Maintainable: It can be maintained, even as your codebase grows. • Portable: It works the same in every environment (operating system) • You have to maintain your code on GitHub.

• You have to keep your GitHub repo public so that anyone can check your code.

• Proper readme file you have to maintain for any project development.

• You should include basic workflow and execution of the entire project in the readme file on GitHub

• Follow the coding standards: <https://www.python.org/dev/peps/pep-0008/>

**Data Source**

The project will require data from the following sources:

-From different different inventories

-product sell prediction and actual sell

**Model Development**

* Choose an appropriate machine learning technique (eg. Logistic Regression, Decision Tree, Random Forest)

**Risks and Mitigation**

- Data Quality: Ensure reliable data sources and handle potential data quality issues through preprocessing.

- Model Accuracy: Continuously evaluate and fine-tune the model to improve demand prediction accuracy.