Ministry of Higher Education and Research
Djilali Bounaama University of Khemis Miliana

Faculty of Materials Science and Computer Science
Department of Artificial Intelligence

H2O.ai Platform Report and Quick Project Summary

Master 1 Artificial Intelligence

Presented by:

Douadjia Abdelkarim
Boudjemaa mohamed amine

H2O.ai Platform Report and Quick Project Summary

1. Overview of H2O.ai Platform

H2O.ai is an open-source machine learning and AI platform designed to automate workflows, simplify model building, and democratize data science. Key features include:

- ✓ Automation: AutoML optimizes and selects the best models automatically.
- ✓ Scalability: Works with large datasets on distributed systems.
- ✓ Flexibility: Supports various ML algorithms, including Gradient Boosting, Random Forest, Deep Learning, and more.
- ✓ Integration: Compatibility with Python, R, and enterprise tools like Spark and Hadoop.

2. Titanic Survival Prediction Project

Objective

To predict passenger survival on the Titanic using H2O AutoML, demonstrating the platform's ease of use and performance in classification tasks.

Dataset Used:

- ✓ Titanic.csv (from Data Science Dojo)
- ✓ Target Variable: Survived
- ✓ Features Used: **Pclass, Sex, Age, SibSp, Parch, Fare, Embarked**

3. Code Walkthrough & Execution

Code Breakdown:

1.Initialize H2O and Load Dataset:

```
import h2o
from h2o.automl import H2OAutoML
h2o.init()
data = h2o.import_file("https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.csv")
```

- ✓ Ensures H2O is running.
- ✓ Loads Titanic dataset from a remote URL.

2.Data Preparation

- ✓ **Dataset:** Titanic passenger data (891 rows, 12 columns).
- ✓ **Preprocessing Steps:**
  - Converted Survived to a categorical feature.
  - Dropped non-predictive columns: **Name, Ticket, Cabin, PassengerId.**
  - Split data into 60% train, 20% validation, and 20% test sets.

```
# Key code snippet:
data['Survived'] = data['Survived'].asfactor()
excluded_columns = ['Survived', 'Name', 'Ticket', 'Cabin', 'PassengerId']
x = [col for col in data.columns if col not in excluded_columns]
y = 'Survived'
```

- ✓ Converts the target variable to categorical format (**asfactor()**).
- ✓ Removes non-informative columns (**Name, Ticket, Cabin, PassengerId**).

3. Splitting Data for Training & Validation:

```
train, valid, test = data.split_frame(ratios=[0.6, 0.2], seed=123)
```

- ✓ Splits data into 60% training, 20% validation, and 20% testing.

4. Running AutoML for Model Selection:

```
aml = H2OAutoML(max_runtime_secs=30, seed=123)
aml.train(x=x, y=y, training_frame=train, validation_frame=valid)
```

- ✓ Runs AutoML for 30 seconds, automatically selecting the best-performing model.

5. Displaying the Leaderboard:

```
print(aml.leaderboard.head())
```

- ✓ Displays the top models sorted by AUC (Area Under Curve).

4. Results & Leaderboard Analysis
Model Performance Metrics:

Model	AUC	Log Loss	RMSE
StackedEnsemble_BestOfFamily	0.847	0.448	0.376
StackedEnsemble_AllModels	0.846	0.450	0.375
GBM_2	0.846	0.448	0.375
XGBoost_1	0.823	0.485	0.396
GLM_1 (Logistic Regression)	0.823	0.483	0.395

Key Takeaways:

- Stacked Ensembles (combining multiple models) achieved the highest AUC score.
- Gradient Boosting (GBM) and XGBoost performed well.
- Logistic Regression (GLM_1), though simple, still provided competitive results.

Key Insight:
Stacked ensembles outperformed individual models, showcasing H2O's strength in combining models for robustness.

6. Strengths of H2O.ai Demonstrated

1. **Speed:** Trained multiple models in 30 seconds.
2. **Automation:** AutoML handled feature engineering, algorithm selection, and stacking.
3. **Transparency:** Leaderboard provided clear model comparisons.
4. **Scalability:** Handled missing values and categoricals (e.g., Sex, Embarked) seamlessly.

7. Limitations

- ✓ Short runtime (30s) limited model depth. Extend to 5–10 minutes for better results.
- ✓ Validation frame usage warnings (cross-validation was still enabled).

Conclusion

H2O.ai bridges the gap between theoretical machine learning and practical implementation. By automating tedious tasks and delivering high-performing models "out of the box," it empowers organizations to focus on deriving value from data rather than wrestling with technical complexity. This Titanic project exemplifies how H2O.ai can accelerate time-to-insight across industries—from healthcare to finance—while maintaining transparency and scalability. As AI adoption grows, platforms like H2O.ai will play a pivotal role in democratizing data science, enabling businesses of all sizes to harness the power of machine learning without requiring vast resources or expertise.