YAP470 – Analytics Project

Status Report

Predicting Road Accident Risk Using Deep Regression on Synthetic Data

Esmanur Ulu – 231101024

Nehir Tıraş – 231101065

Zeynep Yetkin – 231101042

I. State of Data Collection

The dataset for this project was obtained from the **Kaggle competition: Playground Series – Season 5, Episode 10**.
Both training and testing datasets were successfully collected and verified.

- **Training set:** 517,755 samples
- **Test set:** 172,586 samples
- **Total:** 690,341 records

The dataset consists of both numerical and categorical variables and was synthetically generated using a deep learning model trained on real-world simulated road-accident data.
There are no missing or corrupted values. The data is stored and processed using Google Colab for GPU-enabled training and scalability.

Data cleaning, outlier handling, feature encoding, and scaling have been planned based on our 10-step EDA. The data is now ready for the preprocessing and modeling stage.

II. Exploratory Data Analysis (EDA)

A comprehensive 10-step Exploratory Data Analysis (EDA) was conducted on the 517,755-sample training set to identify complex patterns, validate feature importance, and justify our modeling approach. The analysis went beyond standard bivariate charts to include advanced techniques such as model-based Partial Dependence Plots (PDP) and Cramér's V correlation for categorical features.

Key Findings

1.Data Quality & Target Variable:

- **Data Integrity:** The dataset is 100% complete. Analysis confirmed **zero (0) missing values** and **zero (0) duplicate rows**.
- **Target Distribution:** The target variable accident_risk is **symmetrical** (not skewed), centered around a **mean of 0.35 and a median of 0.34**. This is an ideal distribution for regression modeling.
- **Categorical Cardinality:** All 8 categorical features (e.g., lighting, weather, road_type) are **low-cardinality**, containing only 2 to 4 unique values each.

2.**Feature Relationships & Importance:**

- **Strongest Predictors:** violinplot analysis and PDP analysis confirmed that lighting, weather, and time_of_day are the strongest categorical predictors, showing the clearest separation in risk between their categories (e.g., 'daylight' vs. 'dark').
- **Linear Correlation (Heatmap):** A standard correlation matrix showed that num_reported_accidents (+0.41) and curvature (+0.28) have the highest **positive linear** correlation with accident_risk.
- **Categorical Independence (Cramér's V):** A key advanced finding showed that all categorical features are **almost perfectly independent** of each other. The highest inter-feature correlation (e.g., between lighting and weather) was only **0.05**. This proves that all 8 features provide unique, non-redundant information, justifying their inclusion in the model.

3.Critical Non-Linear Discoveries (Model-Based EDA): The most important findings came from training a baseline RandomForest model and analyzing its "brain" using Partial Dependence Plots. This proved the data's non-linearity and confirmed our choice of a Deep Learning model.

- **The "Threshold" Effect:** The curvature feature does not have a linear impact. The model learned a critical **threshold at 0.5**; risk increases slowly up to this point and then "jumps" significantly, proving a complex, non-linear relationship.
- **The "Saturation" Effect:** The num_reported_accidents feature also showed a non-linear "saturation" point. The model learned that the risk increases significantly from 0 to 2 accidents, but **flattens completely after 2**. The difference in risk between 2 accidents and 7 is negligible.

Planned Methodology & Feature Engineering (Based on EDA)

Our EDA findings directly inform our final data preprocessing and modeling strategy.

1. **Categorical Features:** Based on their low-cardinality, all 8 categorical features will be fed into an Entity Embedding layer within our PyTorch model. This will create dense, trainable vector representations for each category, as detailed in our Core MLP (R2) strategy.
2. **Numerical Features (Transform):** To handle the "saturation effect" discovered in the num_reported_accidents PDP plot, we will first apply a Logarithmic Transform to this feature.
3. **Numerical Features (Scaling):** To handle the extreme outliers in the (now transformed) num_reported_accidents, a standard Min-Max scaler is unsuitable. We will scale all numerical features using StandardScaler (or RobustScaler), which is robust to outliers and required for neural network stability.
4. **Modeling Justification:** The discovery of non-linear "thresholds" and "saturation points" (reported_accidents) proves that a simple linear model will be insufficient. This EDA justifies our choice of a Multi-Layer Perceptron in PyTorch, as it is specifically designed to learn and model these complex, non-linear interactions.

III. Comprehensive Modeling Strategy and Technical Justification

The project will employ three primary modeling approaches, resulting in four distinct experiments designed to establish robust performance comparisons and rigorously justify the Deep Learning effort using evidence from the Exploratory Data Analysis.

III.A. Model 1: Classical Baseline Regression (R0) – Justification for Model Complexity

A simple linear regression model, such as **Ridge Regression** or **Support Vector Regression (SVR)** implemented via *Scikit-learn*, will establish the initial performance floor (R0).
This model, utilizing **Pipeline A** (simple encoding), serves as the baseline against which all complex models are evaluated.

The scientific role of this model is to **validate the necessity of deep learning**.
EDA confirmed that the relationship between predictors and the target variable (*accident_risk*) is **non-linear** and may exhibit **threshold-like behavior**, unfit for a linear function.

Interactions such as *curvy road + foggy weather* amplify risk non-additively — exceeding linear capacity.

Hence, the Linear Regression baseline is expected to perform poorly, thereby **justifying the higher-capacity MLP model**.

III.B. Model 2: Tabular SOTA Benchmark (LightGBM Regressor – R1)

The **LightGBM Regressor** is chosen as the **Tabular State-of-the-Art (SOTA) benchmark (R1)** due to its superior performance and scalability for structured data.
LightGBM's Gradient-Boosted Decision Tree (GBDT) structure captures **irregular patterns** and **feature interactions** typical of tabular problems, providing a demanding reference point for deep models.

Optimization will use **Pipeline A (Label Encoding)** and **Optuna-based hyperparameter tuning**, targeting RMSE ≈ **0.056**, consistent with competitive studies.
This optimized LightGBM defines the **minimum acceptable performance threshold** for the Deep Regression model.

III.C. Model 3: Deep Regression Architecture (PyTorch MLP – R2 and R3) – The Core Model

The project's core contribution is the **Deep Regression Architecture** implemented in *PyTorch*.
This architecture directly addresses EDA findings revealing complex non-linearities and interaction effects.

1. Core MLP Architecture (R2)

The input layer handles heterogeneous data via **Pipeline B**, combining scaled numerical features and **categorical entity embeddings**.
Entity Embeddings provide continuous dense representations of categorical variables, allowing richer learning dynamics.

The MLP will contain **3–5 dense layers** with **GELU activations**, chosen for smooth gradient flow and superior performance compared to ReLU.
The final layer has a **single linear output node** for continuous risk prediction (0–1 range).
Training will minimize **Mean Squared Error (MSE)**, directly aligning with the evaluation metric RMSE, using the **Adam optimizer**.

2. Advanced MLP Model (R3) – “Regularization Cocktail”

To challenge the LightGBM SOTA and control overfitting, the **R3** model integrates multiple regularization mechanisms:

- **Batch Normalization:** stabilizes layer input distributions, improving convergence and model robustness.
- **Dropout (0.2–0.4):** prevents neuron co-adaptation, reducing noise memorization.
- **Weight Decay (L2 Regularization):** penalizes large weights within Adam optimizer to improve generalization and numerical stability.

This enhanced architecture forms the **main experimental contribution**, testing how architecture depth and regularization influence model generalization on synthetic data.

Table 1. Comparison of Proposed Modeling Approaches

Model Type	Example Implementation	Role in Project	Key Preprocessing Requirement	Primary Evaluation Metric
Classical Baseline (R0)	Scikit-learn (e.g., SVR)	Establish minimum viable performance	Simple One-Hot/Label Encoding (Pipeline A)	RMSE
Tabular SOTA Benchmark (R1)	LightGBM Regressor	Industry-standard baseline	Label Encoding (Pipeline A)	RMSE
Deep Regression (R2) Core MLP	PyTorch MLP (TabMlp variant)	Main Deep Learning investigation	Standard Scaling + Entity Embeddings (Pipeline B)	RMSE
Deep Regression (R3) Advanced MLP	PyTorch MLP + Regularization Cocktail	Optimized Deep Model; aims to surpass SOTA	Pipeline B + Hyperparameter Scheduling	RMSE

IV. Planned Methodology and Implementation Plan

Phase	Task	Expected Output
Data Preparation	Cleaning, encoding, scaling	Final ready dataset
EDA	Visual and statistical exploration	EDA Notebook with figures
Modeling	Train R0–R3 models	RMSE comparison across models
Evaluation	Metric visualization, interpretation	Tables, plots, significance tests
Final Report	IEEE-format paper + demo	Submission & presentation (Dec 14)

V. Expected Outcomes

- **Target Metric:** RMSE ≤ 0.056 on validation and test sets
- Deep models (R2–R3) expected to outperform LightGBM (R1)
- Feature-importance and embedding analysis will provide interpretability insights
- Establishes a scalable framework for deep regression on synthetic structured data

---

VI. References

[1] S. Sharma and M. R. Patel, “A Study on Traffic Crash Severity Prediction Using Machine Learning Algorithms,” *International Journal of Transportation Science and Technology*, vol. 11, no. 3, pp. 215–229, 2023.

[2] L. Chen, T. Zhang, and K. Xu, “Neural Network Models for Combined Classification and Regression,” *Pattern Recognition Letters*, vol. 158, pp. 23–30, 2022.

[3] M. Zhou, “Deep Neural Networks for Regression Problems,” *arXiv preprint arXiv:2304.01542*, 2023.

[4]R. K. Verma, “Examples of EDA on Deep Learning Projects,” *Medium/Analytics Vidhya*, 2022.

[5] A. Ng and K. Lakshmanan, “A Holistic Guide to Exploratory Data Analysis (EDA) for Machine Learning and Deep Learning,” *Google AI Education*, 2021.

[6] A. Ivaniuk, "Reconsidering Deep Learning for Tabular Data Problem," GoPubby Blog, Oct 21, 2024.

[7]L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on tabular data?," *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[8] Neuromatch Academy, "Deep Learning, W2D1 - Tutorial 2: Regularization," neuromatch.io, 2024

VII. Files Submitted

- **PDF Report:** YAP470_RoadAccidentRisk.pdf
- **EDA Notebook (with outputs):** Predicting Road Accident Risk using Deep Regression_on_Synthetic_Data_EDA.ipynb