Machine Learning-Based Flood Risk Assessment at Street Level: A Graph Embedding Approach in the Tamanduateí River Basin

André Nogueira Sousa
Applied Computing
National Institute for Space Research - INPE
São José dos Campos, Brazil
andrenogsousa@gmail.com

***Abstract*—Urban flooding poses severe risks to metropolitan infrastructure and safety. While traditional hydrological models are robust, they are often computationally expensive and struggle with real-time street-level granularity. This study presents a static baseline model for street-level flood prediction in the Tamanduateí River Basin, São Paulo, utilizing a Graph Machine Learning approach. We integrate heterogeneous geospatial data—Digital Terrain Models (DTM) and hydrography—into a road network graph. Structural properties of the urban grid are captured via Node2Vec embeddings, which are combined with engineered features (TWI, Slope, Distance to River). A Gradient Boosting model (CatBoost) was optimized using a spatial cross-validation strategy to prevent data leakage, achieving an ROC AUC of 0.69 on an unseen microbasin. This work serves as the structural foundation for a broader thesis aimed at developing Spatio-Temporal Graph Neural Networks (ST-GNNs) for real-time forecasting.**

***Index Terms*—urban flooding, graph embeddings, Node2Vec, spatial cross-validation, CatBoost, Tamanduateí River Basin**

I. Introduction

The intensification of extreme weather events due to climate change, coupled with rapid urbanization, has made flood forecasting a critical challenge for city management, [1]. The Tamanduateí River Basin in São Paulo stands as a prime example of a region suffering from recurrent, high-impact flood events. This basin, highly impermeable due to dense urban development, frequently experiences flash floods that disrupt traffic, damage property, and endanger lives.

The central thesis of this research project is the development of a predictive system capable of issuing flood alerts at the specific *street segment* level. Traditional physics-based hydrological models, while accurate for river flow, often lack the granularity or computational speed required to predict flash floods on specific urban roads in real-time. Conversely, pure data-driven approaches often neglect the spatial connectivity of the urban fabric.

The ultimate objective is to implement Spatio-Temporal Graph Neural Networks (ST-GNNs) to predict not just the location, but the timing and intensity of floods using time-series rainfall data. This advanced architecture requires a sophisticated representation of the city's topology to function effectively.

However, before introducing temporal dynamics, it is crucial to establish a robust representation of the spatial domain. This paper details the development of the **static baseline model**, which focuses on:

1) *Topology:* Representing the city as a graph where flood risks propagate through connected road segments.
2) *Susceptibility:* Identifying which streets are inherently prone to flooding due to static factors (terrain, location, and network structure) before any rain falls.

This increment validates the graph construction pipeline and the embedding generation process, ensuring the data structure is ready for the future integration of temporal tensors in the ST-GNN architecture. By successfully modeling the static susceptibility, we establish a "risk prior" that will significantly enhance the learning capability of future dynamic models.

II. Methodology

A. Study Area and Graph Construction

The study area covers the Tamanduateí River Basin, a critical hydrological zone in the São Paulo Metropolitan Area. We utilized a high-resolution Digital Terrain Model (DTM) to capture micro-topographical features, hydrography layers mapping the river network, and official flood occurrence records from 2019. The flood records serve as the ground truth for our supervised learning task.

The road network was extracted from OpenStreetMap using the *OSMnx* Python package [2] and converted into a graph $G = (V, E)$, where V represents intersections and E represents street segments. This graph representation is vital because water flow in urban environments is often constrained and directed by the street layout. A 1 km buffer was applied to the area of interest to mitigate edge effects during graph traversal algorithms, ensuring that nodes at the boundary of the basin have sufficient neighborhood information for embedding generation.

The resultant graph possess the following characteristics:

- Number of Nodes: 48739
- Number of Edges: 73076 (43598 after dropping the buffer)

- Number of Connected Components: 1
- Graph Density: 0.000062
- Mean Clustering Coefficient: 0.0354

B. Feature Engineering

For every edge $e \in E$, we extracted static geospatial features by overlaying the graph onto raster layers and calculating the mean pixel value along the segment. These features were selected based on their hydrological relevance:

- **Distance to River:** Euclidean distance to the nearest water body. Streets closer to the river are naturally more susceptible to overflow events.
- **Slope:** Terrain steepness derived from DTM. Flatter areas are more prone to pooling water, while steep slopes facilitate runoff but can channel water to lower areas.
- **Topographic Wetness Index (TWI):** Calculated as $\ln(SCA/\tan\beta)$, where SCA is the Specific Catchment Area and β is the local slope, [3]. This index indicates potential water accumulation zones, serving as a powerful proxy for soil moisture saturation and runoff potential.

C. Feature and Label Mapping to Graph Structure

A critical challenge in this work is the integration of continuous geospatial information, represented as raster layers, with the discrete topological structure of the street network graph. This section details the methodology for transferring terrain and hydrological properties to graph edges and assigning flood labels based on historical occurrences.

1) Raster-to-Graph Feature Transfer: The engineered features (TWI, Slope, Distance to River) exist as continuous fields in georeferenced raster format with a spatial resolution inherited from the original DTM. To associate these properties with individual street segments (graph edges), we employed a zonal statistics approach implemented using the `rasterstats` library.

For each edge geometry (LineString), the underlying raster pixels were identified through spatial overlay operations. The **mean value** of all pixels intersecting with the street segment was calculated and assigned as an edge attribute. This aggregation strategy was chosen because:

1) It provides a representative summary of the feature along the entire street length.
2) It smooths localized noise in the raster data.
3) It maintains computational efficiency for large-scale urban networks.

The aggregation process can be formally described as:

$$f_{\text{edge}}(e) = \frac{1}{|P_e|} \sum_{p \in P_e} f_{\text{raster}}(p) \tag{1}$$

where P_e is the set of pixels intersecting edge e, and $f_{\text{raster}}(p)$ is the feature value at pixel p.

All three geospatial features were processed using this methodology, with the `all_touched=True` parameter ensuring that pixels partially overlapping the street geometry were included in the computation.

Visual Illustration: Figures 1 through 6 illustrate the transformation from continuous raster features to discrete graph properties. For each feature (Distance to River, Slope, and TWI), we present the raw raster layer followed by the corresponding graph representation where edge colors represent the aggregated mean values.

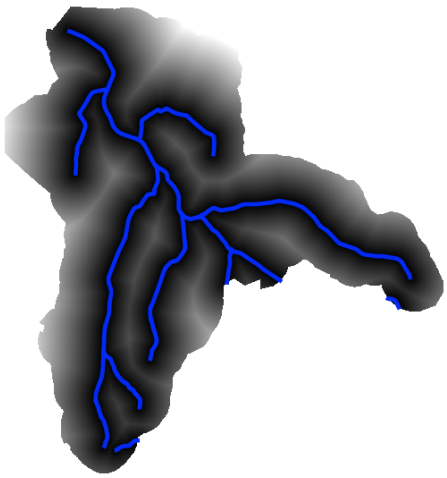

Fig. 1. Distance to River: continuous raster layer showing Euclidean distance from each pixel to the nearest water body.

Fig. 2. Distance to River: mean values assigned to graph edges. Edge colors represent the aggregated distance for each street segment.

2) Flood Label Assignment: The target variable for supervised learning is a binary classification label indicating whether a street segment experienced flooding during the 2019 event inventory. Historical flood occurrences were provided as point geometries with geographic coordinates.

To assign labels to edges, we implemented a proximity-based matching strategy using GeoPandas' `sjoin_nearest` function with a maximum distance threshold of 30 meters. This threshold was selected based on:

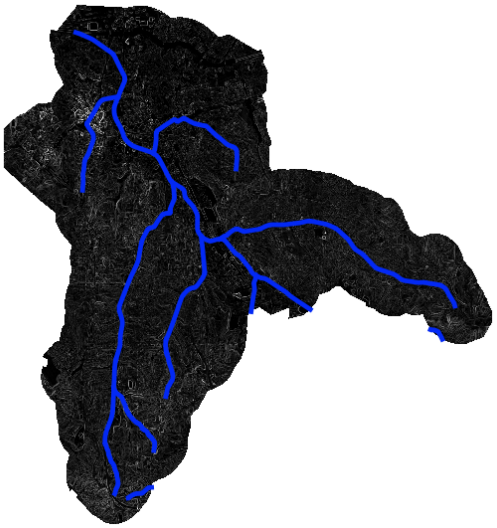

Fig. 3. Slope: continuous raster layer derived from the Digital Terrain Model.

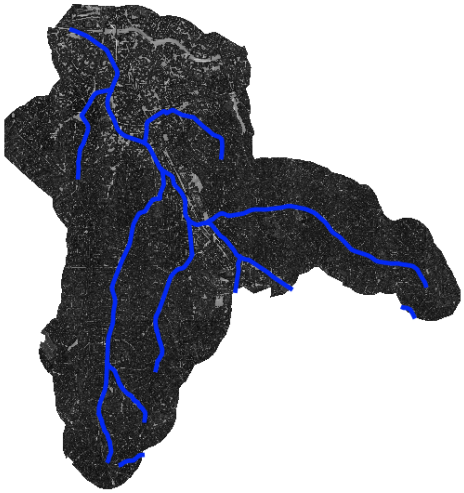

Fig. 5. Topographic Wetness Index (TWI): continuous raster layer indicating water accumulation potential.

Fig. 4. Slope: mean values assigned to graph edges representing terrain steepness along each street.

Fig. 6. Topographic Wetness Index (TWI): mean values assigned to graph edges for flood susceptibility assessment.

- The typical width of urban streets in the study area.
- The need to capture events where flooding occurred near, but not exactly on, the street centerline geometry.

The labeling algorithm operates as follows:

1) For each street edge, identify the nearest flood point.
2) If the distance is $\leq$ 30 meters, mark the edge as a positive class (flood event).
3) If no flood point exists within this radius, the edge is labeled as a negative class (no flood).

To handle cases where multiple flood points were associated with a single street (e.g., a long avenue with multiple reports), we applied a **max aggregation** strategy. If any flood point matched the edge within the threshold, the entire segment was labeled positive. This conservative approach ensures that flood-prone areas are not inadvertently excluded due to spatial discretization artifacts.

The resulting label distribution is highly imbalanced, with flood events representing only 3.65% of the total dataset. This imbalance was explicitly addressed during model training through class weighting strategies (Section IV).

Visual Illustration: Figures 7 and 8 show the labeling process. Figure 7 displays the raw flood point locations overlaid on the street network, while Figure 8 highlights edges that were successfully labeled as positive (red) based on the proximity criterion.

D. Graph Representation Learning (Node2Vec)

To capture the complex topology of the urban grid—which dictates water flow and connectivity—we employed

Fig. 7. Historical flood points (orange) from 2019 event inventory overlaid on the street network graph.

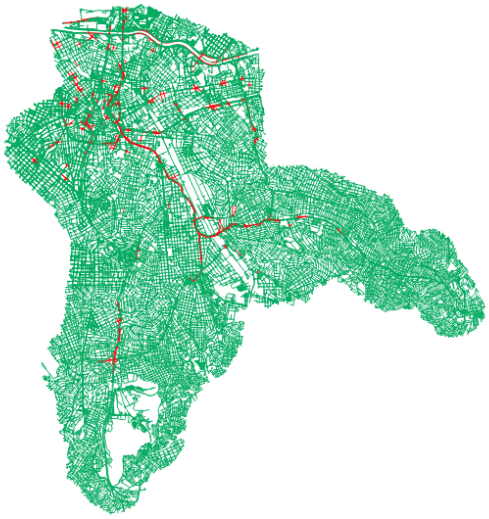

Fig. 8. Street edges labeled as flood-prone (red) based on 30m proximity threshold to historical flood occurrences.

Node2Vec, [4]. This algorithm learns low-dimensional representations (embeddings) for nodes by optimizing a neighborhood preserving objective via random walks. Unlike simple adjacency matrices, embeddings capture higher-order structural similarities (e.g., identifying streets that act as major connectors or sinkholes).

Based on our implementation using `networkx` and `node2vec` libraries, the following hyperparameters were selected:

- **Dimensions:** 64 (Vector size). This dimensionality offers a balance between encoding sufficient structural information and maintaining computational efficiency.
- **Walk Length:** 30 nodes per walk. This length allows the algorithm to explore the local neighborhood structure deeply.
- **Num Walks:** 10 random walks per node. Multiple walks ensure a robust sampling of the graph structure.
- **Parameters p and q:** Set to 1.0. This configuration corresponds to an unbiased random walk, effectively balancing the exploration of local clusters (Breadth-First Search behavior) and the discovery of macro-structural roles (Depth-First Search behavior).

Since our prediction target is on the *street* (edge) rather than the intersection (node), we computed edge embeddings by aggregating the source (u) and target (v) node vectors. We utilized the **Average** operator:

$$f(u, v) = \frac{\text{Emb}(u) + \text{Emb}(v)}{2} \tag{2}$$

This resulted in a feature vector of 64 dimensions for each street segment, which was then concatenated with the physical features (TWI, Slope, Distance) to form the final input vector for the classifier.

III. Experimental Setup

A. Data Splitting Strategy

A critical challenge in geospatial Machine Learning is *Spatial Autocorrelation*, [6]. Features and labels of geographically adjacent streets are highly correlated. Randomly splitting streets into train/test sets would cause "data leakage," where the model could simply memorize the label of a neighbor rather than learning the underlying risk factors.

To address this, we implemented a *Geographic Split* based on hydrological Microbasins (see Fig. 9).

- **Test Set (Hold-out):** Microbasin 'AC 1.1.12' was explicitly selected for testing. This basin exhibits a high flood rate of 6.09%, making it a rigorous stress test for the model's ability to generalize to high-risk unseen areas.
- **Training Set:** All other microbasins (83.55% of data).

B. Spatial Cross-Validation

Within the training set, we employed a `GroupKFold` strategy ($k = 3$). The "groups" were defined by Microbasin IDs. This ensures that in every validation step, the model learns from a set of basins and predicts on entirely different basins. This simulates a realistic deploy where the model must generalize to new neighborhoods or sub-basins that were not part of the training data.

Table I presents the data volume distribution in the train folds and in the test data.

TABLE I
Dataset distribution across test and cross-validation folds.

Group	Streets	Flood Rate	% Data	% Flood
Test (Hold-out)	**7,172**	**6.09%**	**16.45%**	**27.45%**
Fold 1	11,802	3.76%	27.07%	27.89%
Fold 2	13,385	4.51%	30.70%	37.94%
Fold 3	11,239	0.95%	25.78%	6.72%
Total	**43,598**	**3.65%**	**100%**	**100%**

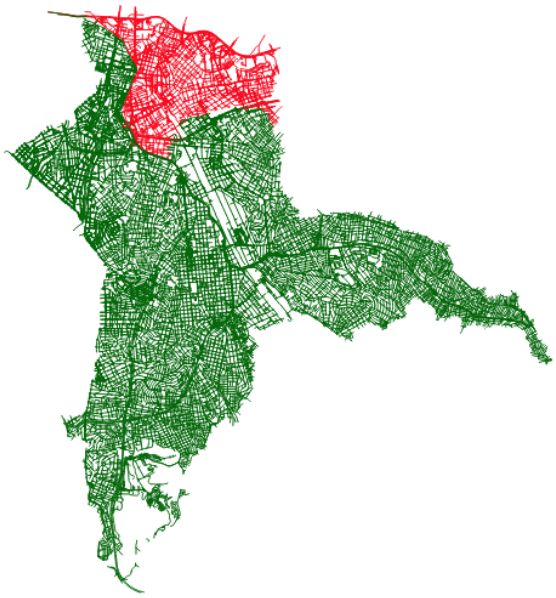

Fig. 9. Geographic split strategy. The red area ('AC 1.1.12') is completely held out during training to test generalization on unseen regions.

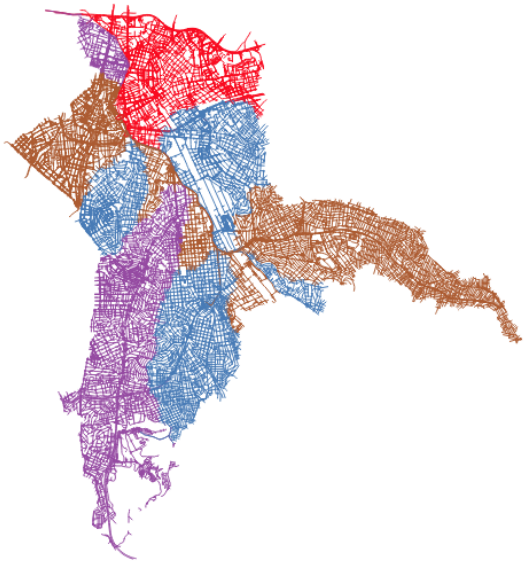

Fig. 10. Distribution of flood events across the 3 Folds and the Test set. Note the varying flood rates (e.g., Fold 2 has higher occurrence), providing a robust stress-test for the model.

C. Model Configuration

We trained a *CatBoostClassifier*, a gradient boosting algorithm chosen for its effective handling of tabular data and robust performance on categorical features (though our features are primarily numerical), [5].

- **Hyperparameter Optimization:** We used **Optuna** ([7]) with 50 trials to maximize ROC AUC. The optimization process searched for the best combination of tree depth, learning rate, and regularization terms.
- **Best Parameters:** The search converged to a tree depth of 10, a learning rate of 0.0147, and 238 iterations. The relatively deep trees suggest the model is capturing complex, non-linear interactions between the topological embeddings and the physical features.
- **Imbalance Handling:** The dataset is highly imbalanced, with non-flood events far outnumbering flood events. The parameter `scale_pos_weight` was dynamically calculated based on the ratio of negative to positive samples in the training set to penalize misclassification of the minority class more heavily.

IV. Results and Discussion

A. Performance Metrics

The model was evaluated on the unseen Test Microbasin 'AC 1.1.12'. The performance metrics were:

- **Accuracy:** 93.57%
- **ROC AUC:** 0.6885
- **F1-Score:** 0.2553

While accuracy is high, this metric can be misleading due to the class imbalance (most streets do not flood). The ROC AUC of ≈ 0.69 is the more critical metric, indicating that the model has learned a meaningful signal regarding flood susceptibility. It distinguishes risky streets from safe ones significantly better than random chance (0.5), validating the hypothesis that static features alone hold predictive power. The low F1-score reflects the inherent difficulty of the task and the trade-off between precision and recall; the model generates false positives, likely identifying streets that could flood given their characteristics, even if they didn't flood in the specific 2019 window.

B. Error Analysis

The Confusion Matrix (Fig. 11) shows that the model is conservative. It correctly identifies the majority of non-flooded streets. The false positives (predicting flood where none occurred) are preferable in a risk alert context—it is safer to warn residents of a potential risk that doesn't materialize than to miss a catastrophic event.

The ROC Curve (Fig. 12) illustrates the trade-off between sensitivity (True Positive Rate) and specificity (1 - False Positive Rate). The curve consistently stays above the diagonal, confirming the model's predictive capability across different decision thresholds.

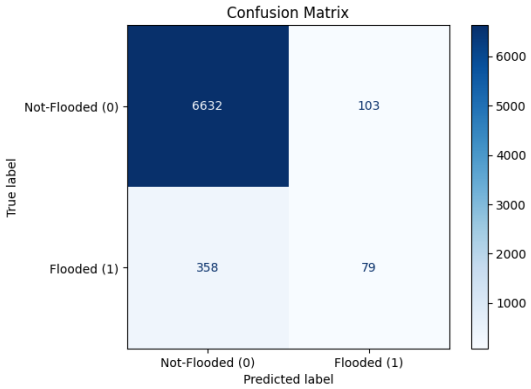

Fig. 11. Confusion Matrix on the Test Set. The model successfully identifies a significant portion of non-flooded areas but faces challenges with the minority positive class.

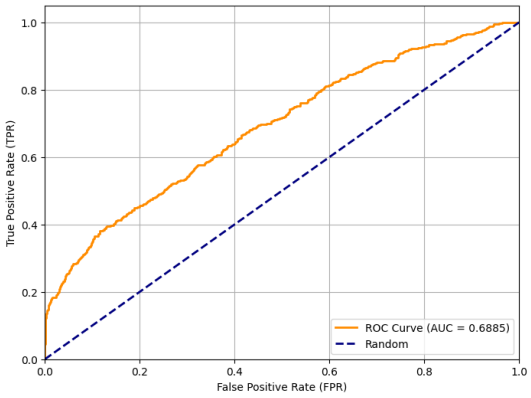

Fig. 12. ROC Curve for the Test Set (AUC = 0.6885). The curve above the diagonal confirms predictive power on unseen geographical data.

C. Feature Importance

The feature importance analysis (Fig. 13) validates the hybrid approach of combining physical features with learned graph embeddings:

1) **Dominance of Hydrology:** `mean_dist_river` (Distance to River) is by far the most important predictor ($\approx 16.5\%$). This aligns with basic hydrological principles, as proximity to the main channel is a primary risk factor for fluvial flooding.
2) **Role of Topology:** Graph embeddings (specifically dimensions `emb_5`, `emb_17`, `emb_12`) occupy top positions in the importance ranking. This is a significant finding. It proves that the *structural context* of a street within the network (captured unsupervised by Node2Vec) adds unique information that local physical features (like Slope) do not capture alone. These embeddings likely encode information about the street's connectivity, centrality, or its role in the wider urban valley, which are crucial for understanding water accumulation pathways.

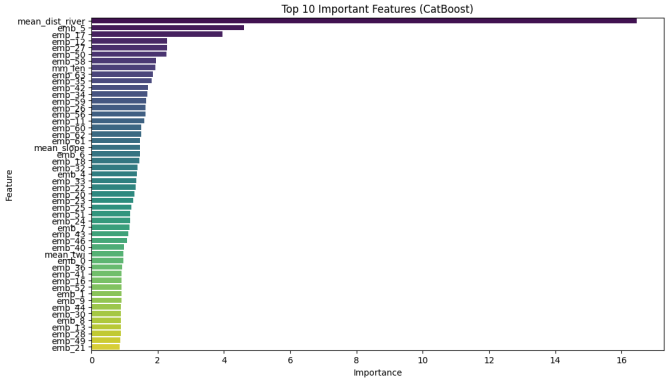

Fig. 13. Feature Importance. Distance to river is dominant, but abstract embedding features (`emb_X`) contribute significantly to the decision trees.

V. Conclusion and Future Work

This study successfully established a static baseline for street-level flood prediction in the Tamanduateí basin. We demonstrated that combining physical terrain features with topological graph embeddings improves the representation of flood susceptibility compared to using physical features alone. Crucially, the use of spatial cross-validation ensures the reliability of our metrics on unseen geographic areas, confirming the model's ability to generalize.

The ROC AUC of 0.6885 represents a solid foundation. It confirms that the graph construction and embedding generation pipelines are robust and capturing relevant spatial signals. The identification of graph embeddings as top-tier features validates the Graph Machine Learning approach for this domain.

A. Future Work

The static model serves as the encoder backbone for the next phase of the thesis, which will transition from susceptibility mapping to real-time forecasting:

1) **Temporal Integration:** We will introduce high-resolution rainfall time series data. These will be modeled as dynamic node features, adding a time dimension to our graph.
2) **ST-GNN Implementation:** The static graph structure constructed here will be processed by Spatio-Temporal Graph Neural Networks (e.g., A3T-GCN or Temporal Graph Convolutional Networks). These advanced models will leverage both the static "risk prior" established in this work and the dynamic rainfall data to predict flood probability T minutes into the future.
3) **Dynamic Embeddings:** We will explore evolving graph structures or dynamic embeddings to account for changing urban conditions or road closures during flood events.
4) **Edge Feature Refinement:** The current aggregation method (mean value) for long streets is a simplification. Future work will evaluate multiple sampling points per edge or more sophisticated spatial aggregation methods.

5) **Advanced tuning:** In the current project little attention was given to parameters related to the embedding generation on *node2vec*, different algorithms might be tested along with tuning of the embedding algorithm parameters.
6) **Graph Based Features:** Beyond embeddings, we can also calculate graph node metrics, such as centrality, betweeness and so on, to be used as features in the ML training.

REFERENCES

[1] Schreider et al., "Climate Change Impacts on Urban Flooding," Climatic Change 47, 91–115 (2000). https://doi.org/10.1023/A:1005621523177.

[2] Boeing, G., OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. Computers, Environment and Urban Systems, vol. 65, pp. 126-139, 2017.

[3] Beven, K. J. and Kirkby, M. J., A physically based, variable contributing area model of basin hydrology. Hydrological Sciences Bulletin, vol. 24, no. 1, pp. 43-69, 1979.

[4] Grover, A. and Leskovec, J., node2vec: Scalable Feature Learning for Networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855-864, 2016.

[5] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A., CatBoost: unbiased boosting with categorical features. In https://doi.org/10.48550/arXiv.1706.09516, 2018.

[6] Roberts, D. R., et al., Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. Ecography, vol. 40, no. 8, pp. 913-929, 2017.

[7] Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M., Optuna: A Next-generation Hyperparameter Optimization Framework. In https://doi.org/10.48550/arXiv.1907.10902, 2019.