Proceedings of Machine Learning Research 266:1–12, 2025 Conformal and Probabilistic Prediction with Applications

Testing by Betting for Anomaly Detection in Rental E-Scooter GNSS Traces

Marco Capuccini MARCO.CAPUCCINI@VOIAPP.IO
Voi Technology AB, Stockholm, Sweden
Division of Transport Planning, KTH Royal Institute of Technology, Stockholm, Sweden

Rahul Rajendra Pai RAHUL.PAI@CHALMERS.SE
Division of Vehicle Safety, Chalmers University of Technology, Gothenburg, Sweden

Lars Carlsson LARS.CARLSSON@JU.SE
Department of Computing, Jönköping University, Jönköping, Sweden

Editor: Khuong An Nguyen, Zhiyuan Luo, Harris Papadopoulos, Tuwe Löfström, Lars Carlsson and Henrik Boström

Abstract

Shared micromobility, particularly rental e-scooters, has rapidly transformed urban transportation. Voi Technology has been at the forefront of this shift, powering over 300M rides across Europe. While most customers use the service responsibly, mitigating reckless riding emerges as a significant challenge given the high ridership. Previous research shows that riders taking indirect routes are more likely to be involved in safety-critical events, suggesting potentially irresponsible riding behavior. However, directness alone can overlook important intra-trip patterns. Therefore, in this study, we use GNSS positioning as a proxy for intra-trip riding behavior. We model a typical ride as a sequence of turning angles derived from GNSS coordinates and detect anomalies leveraging the testing-by-betting framework, which provides formal guarantees on false positive rates while achieving a favorable trade-off with false negatives. The presented method is designed to operate under limited onboard compute, with minimal complexity for deployment across large vehicle fleets, without requiring the GNSS trace to be stored—a key privacy advantage. In a real-world evaluation, the method detects approximately 60% of reckless rides while maintaining operationally acceptable false positive rates.

Keywords: micromobility, safety, testing by betting, martingale testing

1. Introduction

Shared micromobility, particularly rental e-scooters, has experienced exponential growth in recent years, transforming urban transportation at a pace unmatched by previous mobility modes. This rapid adoption is exemplified by the 257M trips taken on shared e-scooters across Europe in 2024 alone (Fluctuo, 2025). Voi Technology has been part of this transformation from the start, powering over 300M rides since 2018 across roughly 100 cities in 12 European countries. With hundreds of thousands of trips per day from millions of riders, Voi collects unique insights that enable the development of tools to make micromobility safer—an increasingly pressing challenge as ridership continues to grow (Stigson et al., 2021).

Previous research has explored various factors that contribute to safety risks in e-scooter use. Notably, Pai and Dozza (2025) found that rides taking indirect routes (i.e., those with detours) are five times more likely to be associated with safety-critical events than those following more direct paths. While this highlights the relevance of directness as a trip-level feature, such a metric can overlook important behavioral patterns that occur within a ride; riders may follow a direct route but ride erratically, or take a longer path while maintaining safe behavior. Moreover, directness and similar trip-level features are only available once a ride has ended, limiting our ability to act on potentially unsafe behavior in real time.

In this study, we focus on detecting anomalous rides based on intra-trip behavior, rather than relying on trip-level features such as directness. To this end, we leverage data from the Global Navigation Satellite System (GNSS), a satellite-based positioning technology, which is available across Voi's e-scooter fleet. In particular, GNSS traces provide a lightweight source of information on how a ride unfolds over time, making them a suitable proxy for intra-trip riding behavior in an online detection setting.

For operational adoption, a positioning-based anomaly detection algorithm needs to satisfy a few requirements. First, while the goal is to detect and mitigate significant erratic behavior, maintaining a good user experience demands strict control over the false positive rate to prevent unnecessary intervention. Moreover, to enable real-time intervention and enhance privacy, the algorithm is preferably run on the vehicle, under limited processing power. Finally, a response should be triggered as soon as enough evidence is collected, allowing us to, for example, show a message, reduce speed, or interrupt a ride.

The testing-by-betting framework introduced by Shafer and Vovk (Shafer, 2021; Shafer and Vovk, 2019) offers a well-suited basis for developing an algorithm that meets all of the aforementioned constraints—providing theoretically grounded control of the false positive rate, efficient computation, and responsiveness in an online setting.

To this end, the contributions of this work are:

- We formalize the task of online anomaly detection in GNSS traces using the testing-by-betting framework.
- We present a practical implementation suitable for real-time evaluation under constrained computational resources.
- We empirically evaluate the method on GNSS traces from one of Voi's markets, demonstrating both controlled false positive rates and effective detection of potentially reckless rides.

The remainder of this paper is structured as follows. Section 2 presents the method in detail, including the theoretical foundation and a practical implementation. Section 3 describes the evaluation setup and reports results on both validity and effectiveness. Section 4 discusses related work. Finally, Section 5 ends the paper with a discussion and concluding remarks.

2. Method

The aim of the presented method is to detect anomalies in GNSS traces while ensuring a valid upper bound on the false-positive rate. GNSS traces consist of a series of time-ordered coordinates, i.e., latitude and longitude, sampled at a consistent rate. For each ride, these traces represent the path of an individual e-scooter over time. For the purposes of this study, we assume that an anomalous ride is characterized by a path that exhibits significantly more directional changes than usual, reflecting a higher degree of irregularity. An example of such a ride is shown in Figure 1.

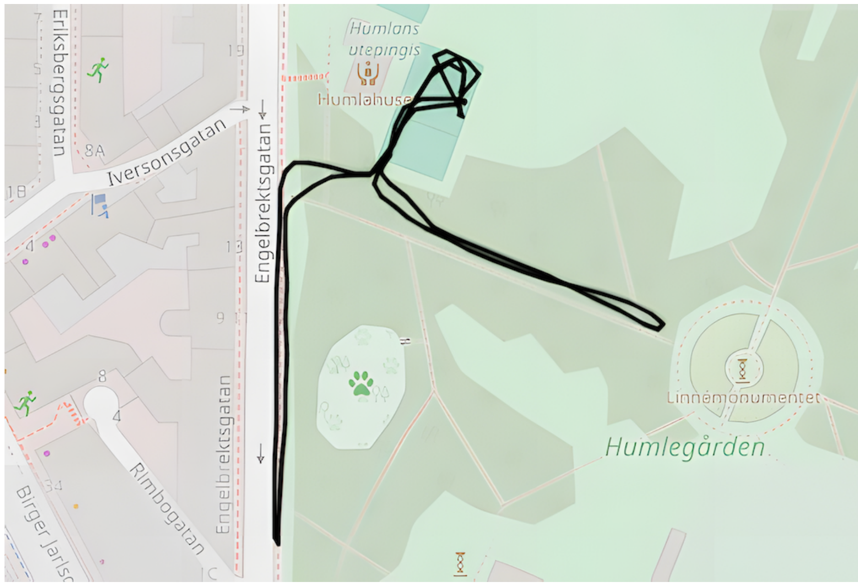

Figure 1: Anomalous ride with excessive directional changes, which may be indicative of reckless riding behavior.

Since our goal is to detect excessive directional changes, we model rides as sequences of turning angles over time, ranging from $-180°$ to $+180°$. These angles are calculated using segments defined by consecutive latitude and longitude pairs. Using this model, we estimate the distribution of turning angles for a typical ride, thereby allowing us to identify anomalous rides whose turning behavior is unlikely to originate from that distribution, subject to a specified significance level—as reflected in the broader, more dispersed histogram shown in Figure 2. Finally, to enable a timely response, the test is preferably run online, allowing an anomaly to be flagged as soon as sufficient evidence is collected.

The remainder of this section further details the method and is structured as follows:

1. We outline an online betting test designed to detect anomalies in turning behavior.

2. We describe the procedure for estimating the probability density function (PDF) of turning angles.

3. We describe how turning angles are computed from GNSS traces.

2.1. Online Betting Test for Anomaly Detection in Turning Behavior

The presented anomaly detection method is based on the testing-by-betting framework developed by Shafer and Vovk (Shafer, 2021; Shafer and Vovk, 2019).

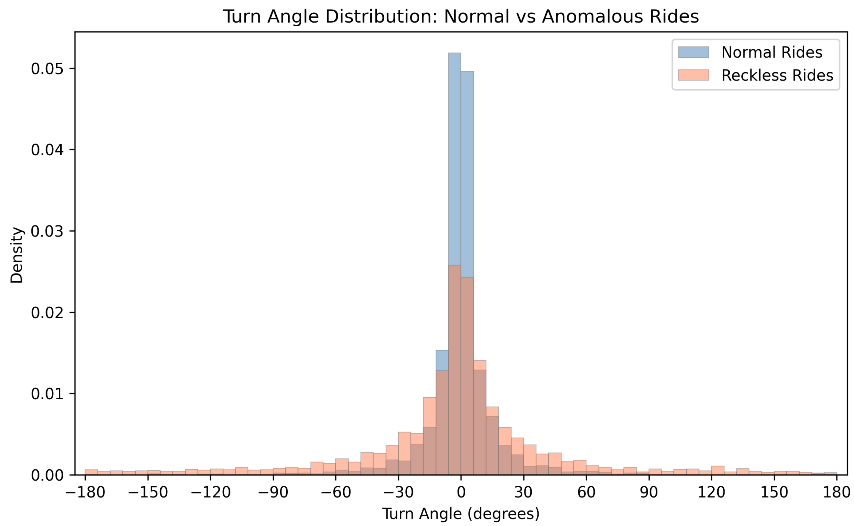

Figure 2: Histogram of turning angle distributions for 10 normal and 10 anomalous rides, as labeled by three human annotators. Anomalous rides show more frequent and extreme directional changes. See Section 2.3 for details on turning angle computation.

2.1.1. Theoretical foundation

Given a probability distribution P that describes turning behavior during a typical ride, and a sequence of turning angles $s_1, s_2, \ldots, s_n$ from a ride we want to evaluate, we can frame anomaly detection as a betting game in which we gain betting capital if $s_1, s_2, \ldots, s_n$ do not originate from P. In an online setting, we start with an initial budget $S_0 = 1$. For each of the observed angles, we risk the entire budget and the return is determined by a scoring function f, which represents the factor by which our bet is multiplied. That is, the budget evolves according to

$$S_{i+1} = S_i \cdot f(s_{i+1}), \tag{1}$$

for $i = 0, 1, \ldots, n-1$.

We want the budget to grow if $s_1, s_2, \ldots, s_n$ do not originate from P and to decrease otherwise. Therefore, we can define f as the following likelihood ratio

$$f(s) = \frac{u(s)}{p(s)}, \tag{2}$$

where u is the PDF of the uniform distribution, i.e., $u(s) = 1/360$ for $-180 \leq s \leq 180$ and $u(s) = 0$ otherwise, and p is the PDF of P. Intuitively, $f(s) > 1$ if s is more likely to be sampled at random than coming from P, leading to a budget increase.

We now show that, with a scoring function defined as in 2, the sequence $S_0, S_1, S_2, \ldots$ forms a non-negative martingale when $s_1, s_2, \ldots$ originate from P. That is, $S_i \geq 0$ and $E(S_{i+1} \mid S_1, \ldots, S_i) = S_i$ for all $i = 0, 1, 2, \ldots$. The non-negativity holds because the likelihood ratio defining the scoring function is always non-negative, and $S_0 = 1$ by definition. The martingale property follows from

$$E(S_{i+1} \mid S_1, \ldots, S_i) = \int \left(\prod_{j=1}^{i} \frac{u(s_j)}{p(s_j)} \right) \frac{u(s)}{p(s)} p(s)\, ds$$

$$= \left(\prod_{i=1}^{i} \frac{u(s_j)}{p(s_j)} \right) \int \frac{u(s)}{p(s)} p(s)\, ds$$

$$= S_i \int \frac{u(s)}{\cancel{p(s)}} \cancel{p(s)}\, ds$$

$$= S_i \int u(s)\, ds = S_i.$$

It directly follows that the sequence $S_0, S_1, S_2, \ldots$ forms a non-negative supermartingale, i.e., $S_i \geq 0$ and $E(S_{i+1} \mid S_1, \ldots, S_i) \leq S_i$ for all $i = 0, 1, 2, \ldots$. Under this condition, Ville's inequality (Ville, 1939) implies that the probability of the budget ever reaching or exceeding a constant C is at most $1/C$. Formally

$$P\left(\exists n : S_n \geq C\right) \leq \frac{1}{C}, \quad \forall C > 0.$$

In the context of our anomaly detection algorithm, we can take advantage of this property to define a valid significance level, which also corresponds to an upper bound on the false positive rate. Specifically, given a significance level σ, for each $s_1, s_2, \ldots, s_n$ we update the budget according to 1. If during this process the budget ever reaches or exceeds $1/\sigma$, we signal an anomaly. Conversely, if $S_i < 1/\sigma$ for $i = 1, 2, \ldots n$ there is no anomaly detected.

2.1.2. Implementation

For a practical implementation, it is important to account for numerical stability, as well as temporal dependence among $s_1, s_2, \ldots, s_n$, which are ordered in time.

For the typical ride, the likelihood ratio defined by the scoring function will often be close to 0. Therefore, to prevent the budget from reaching zero due to underflow, making recovery impossible, we can implement the update as

$$S_{i+1} = \max\left(\varepsilon, S_i \cdot \frac{u(s)}{p(s)}\right), \tag{3}$$

for $i = 0, 1, \ldots, n-1$, where $\varepsilon > 0$ is a small constant. Another option is to work in the logarithmic space, that is

$$S_{i+1} = S_i + \log u(s) - \log p(s), \;\; S_0 = 0, \tag{4}$$

for $i = 0, 1, \ldots, n-1$, which requires exponentiation of the budget before testing Ville's inequality. In contrast to the former, which forms a submartingale—Doob's inequality applies Doob (1953)—this budget update maintains the supermartingale property, and it is better at handling small likelihood ratios. However, as it does not put any bound on how much the budget can decrease, it can make it harder for it to recover.

Next, to address the temporal dependence, in $s_1, s_2, \ldots, s_n$, we introduce a sliding window of fixed size K. The window advances in steps of K, and within each window, we shuffle the observed angles before updating the budget corresponding to that batch. The window size K is chosen to be large enough to eliminate the effects of temporal dependencies. While this is a simple approach, it introduces a detection delay of at most K time steps.

In summary, Algorithm 1 shows the practical implementation of the presented anomaly detection method.

Algorithm 1 Practical Implementation

```
Input: Sequence of turning angles s_1, s_2, ..., s_n; reference PDF p; window size K; significance level σ
Output: True if the anomaly is detected, False otherwise
Initialize budget S ← 1 (or log S ← 0 in log space)
Set threshold C ← 1/σ
for t ∈ {0, K, 2K, ..., ⌊(n − 1)/K⌋ · K} do
    Define window W ← {s_{t+1}, s_{t+2}, ..., s_{min(t+K,n)}}
    Shuffle elements in W
    foreach s ∈ W do
        Update S according to s using 3 (or 4 in log space)
        if S ≥ C (or exp S ≥ C in log space) then
            return True
        end
    end
end
return False
```

We discuss the complexity of Algorithm 1 in conjunction with the PDF estimator presented in the following subsection.

2.2. Estimating the PDF of Turning Angles

Algorithm 1 bets against a reference PDF p, which reflects the distribution of turning angles from a representative set of typical rides. We chose to estimate this PDF using a histogram-based approach for its simplicity, interpretability, and computational efficiency.

To construct the reference histogram, we first collect a sufficiently large sample of turning angles from typical rides. The full range of turning angles, from $-180°$ to $180°$, is then divided into m equally spaced bins, each with a fixed width w. Let h_j be the number of samples falling into bin j, for $j = 1, \ldots, m$, and let l be the total number of samples used to construct the histogram. We approximate the likelihood of a new turning angle s falling into bin j as

$$p(s) \approx \frac{h_j}{l \cdot w}.$$

This yields a piecewise constant approximation of the true PDF, where each bin contributes a flat density value proportional to its sample count.

Locating the corresponding bin for a given turning angle requires $O(\log m)$ time. Therefore, the overall complexity of Algorithm 1 is $O(n \log m)$; recall that n is the number of turning angles in the ride being evaluated.

2.3. Computing Turning Angles form GNSS traces

Given a series of consecutive latitude and longitude pairs sampled at a consistent rate, we define a sequence of segments, each formed by two successive coordinate pairs. Since the coordinates lie on a geoid, we compute the forward azimuth for each segment, defined as the clockwise angle from true north to the direction of the segment (Vincenty, 1975). Then, for two consecutive azimuths A_i and A_{i+1}, we compute the turning angle as

$$(A_{i+1} - A_i + 180) \bmod 360 - 180,$$

that is, the smallest signed angle of rotation needed to turn from the direction of one segment to the next. The result lies in the interval $[-180°, +180°]$, where negative values indicate left turns and positive values indicate right turns.

3. Evaluation

We empirically evaluate the proposed method in terms of both validity, by ensuring that the false positive rate does not exceed the specified significance level, and effectiveness in detecting potentially reckless rides using a real-world dataset.

3.1. Validity Assessment

To validate our method in terms of false positive rate guarantees, we sampled 1K rides from the beginning of 2025, each with a minimum duration of 10 minutes, from one of Voi's markets. Because anomalous turning behavior is rare, this relatively small sample is unlikely to contain anomalies, making it suitable for the presented evaluation. For each ride, GNSS coordinates were sampled every 2 seconds, ensuring sufficiently spaced segments to compute turning angles.

From this dataset, we used 800 rides to compute the reference PDF p, using a bin width $w = 360$. The remaining 200 rides were used for evaluation at a significance level of 0.01, i.e., with a decision threshold $C = 100$. To account for statistical fluctuations, we repeated the experiment 100 times with different splits, and tested window sizes $K \in \{30, 60, 90, 120\}$, corresponding to 1, 2, 3, and 4 minutes of ride time. The results are presented in Table 1 and Table 2, corresponding to the epsilon-bounded multiplicative update (see 3, $\epsilon = 2.22 \times 10^{-16}$) and the log-space additive update (see 4), respectively.

Table 1: False Positive Rates (Epsilon-Bounded Multiplicative Update)

Window Size	**Mean**	**Min**	**25%**	**50%**	**75%**	**Max**
30	8.78	2	7.00	9.00	10.00	17
60	4.16	1	3.00	4.00	5.00	9
90	1.76	0	1.00	2.00	2.25	6
120	1.72	0	1.00	2.00	2.00	6

Table 2: False Positive Rates (Log-Space Additive Update)

Window Size	**Mean**	**Min**	**25%**	**50%**	**75%**	**Max**
30	8.38	2	6.75	8.00	10.0	16
60	3.98	1	3.00	4.00	5.00	9
90	1.57	0	1.00	1.00	2.00	5
120	1.49	0	1.00	1.00	2.00	5

The empirical evidence is consistent with the method's theoretical validity for window sizes $K \geq 90$, where the mean number of false positives remains below 2 out of 200 rides, consistent with the specified significance level $\sigma = 0.01$. This indicates that a window size of 90 is likely sufficient to eliminate temporal dependencies in the sequence of turning angles. Moreover, while both the epsilon-bounded and log-space update variants maintain validity, the log-space implementation appears slightly more conservative, yielding consistently lower false positive rates across all window sizes. This is likely due to the log-space update allowing for an unbounded budget decrease, making it harder to recover from a very small value.

3.2. Effectiveness Assessment

To evaluate the extent to which the proposed method achieves its goal, i.e., flagging potentially reckless rides, we construct a dataset using real-world data from one of Voi's markets. Specifically, we sample rides from the beginning of 2025, focusing on cases where accelerometer and gyroscope readings indicate that the e-scooter may have fallen over at speed; while this may introduce a slight bias, it increases the chance of including reckless riding in the dataset. Next, we present the corresponding GNSS traces, overlaid on a map, to three human agents with at least one year of experience in the shared e-scooter mobility space. The agents have the option to label rides as anomalous, i.e., likely exhibiting reckless behavior, or as normal. Additionally, they may label rides as likely affected by GNSS noise or as uncertain if they are unsure about the nature of the riding behavior. Finally, to ensure robust evaluation, we consider rides for which all three agents agreed on the label with certainty; cases marked as likely affected by noise are excluded, as noise handling is beyond the scope of this study.

The resulting dataset comprises 627 rides, of which 21 are labeled as anomalous. We evaluate it using the presented method with a window size $K = 90$, a reference PDF p estimated from 1,000 rides sampled at random from the same market (using a bin width of 360), and a threshold $C = 10{,}000$; this choice is appropriate given the planned deployment scale, as Voi operates hundreds of thousands of rides per day across all markets combined. As a baseline, we use a heuristic previously employed by Pai and Dozza (2025). For each ride, we first compute its directness factor—i.e., the ratio of the recommended distance by OpenStreetMap to the actual distance of the trip. Then, if the directness factor is lower than 0.6, we flag the ride as anomalous.

The results are summarized in Table 3, Table 4, and Table 5, showing the confusion matrices for the epsilon-bounded, log-space, and directness heuristic variants, respectively. For the anomalous class, the epsilon-bounded variant achieves a precision of 72% and a recall of 62%, while the log-space variant yields a slightly higher precision of 75% but a lower recall of 57%. Overall, the two variants perform comparably, with the epsilon-bounded update offering improved recall at the cost of a modest reduction in precision. It is important to note that formal validity is not preserved in this analysis, as the objective is to detect rides exhibiting potentially reckless behavior rather than to identify out-of-distribution turning patterns. The directness heuristic, by contrast, achieves a high recall of 95% at the cost of an operationally unacceptable precision of 38%, favoring sensitivity to anomalous behavior over predictive accuracy. This is due to the fact that the heuristic does not account for intra-trip patterns, which are instead captured by turning angles.

Table 3: Confusion Matrix (Epsilon-Bounded Multiplicative Update)

	Predicted Normal	**Predicted Anomaly**
Actual Normal	601 (99.2%)	5 (0.80%)
Actual Anomalous	8 (38.1%)	13 (61.9%)

Table 4: Confusion Matrix (Log-Space Additive Update)

	Predicted Normal	**Predicted Anomaly**
Actual Normal	602 (99.3%)	4 (0.70%)
Actual Anomalous	9 (42.9%)	12 (57.1%)

Table 5: Confusion Matrix (Directness Heuristic)

	Predicted Normal	**Predicted Anomaly**
Actual Normal	573 (94.6%)	33 (5.45%)
Actual Anomalous	1 (4.76%)	20 (95.2%)

4. Related Work

Anomaly detection in mobility research often relies on Machine Learning (ML) methods. While these methods are rigorously tested, they generally offer no theoretical guarantees on false positive rates. To this end, Hansson and Congreve Lifh (2022), in collaboration with Voi, applied Isolation Forest (Liu et al., 2008) and autoencoders (Beggel et al., 2020) to detect anomalies in sensor data collected during rides. In particular, they used accelerometer and gyroscope readings, as well as velocity measured both at the wheel and via the GNSS system. For training and evaluation, they collected one day of naturalistic data (i.e., produced by end-user riding) from one of Voi's markets, as well as data from simulated reckless riding in a controlled setting. Conversely, the work presented in this paper solely focuses on naturalistic data, collected over a larger time span.

Yaqoob et al. (2023) focused on data collected from bicycles equipped with a global positioning system device. Like the aforementioned study, they used an autoencoder to detect anomalies. As in the work presented in this paper, their focus included identifying safety-threatening events such as near misses. However, their model was trained and evaluated on a relatively small scale—10 bicycles covering a 4 km course—collected in a fairly controlled setting, again non-naturalistic compared to the data used in this study.

Given that the method introduced in this paper is grounded in testing-by-betting, it is important to mention the martingale-testing framework introduced by Vovk et al. (2003), which enables online exchangeability testing with theory-grounded false positive control. In related work, martingale-testing was employed by Ho et al. (2019) to detect drift in flight behavior and by Cherubin et al. (2018) for feature selection in an anomaly detection setting. Notably, martingale-testing is based on Conformal Prediction (CP) (Vovk et al., 2005) and requires the computation of non-conformity scores, which are typically derived from an underlying ML model. It is also worth noting that CP can be used directly to build a well-calibrated anomaly detector, as shown by Laxhammar and Falkman (2015). However, while martingale-testing detects anomalies as data points that are not exchangeable with the reference data, CP requires reference data to be exchangeable with the evaluated data for the theoretical false positive guarantees to hold.

The method presented in this paper differs from the reviewed work by not relying on an underlying ML model. While ML-based methods may improve detection performance, this design choice reflects the practical constraints of vehicle-side deployment. These constraints are not only computational but also relate to the operational complexity of running and maintaining ML models across a large vehicle fleet.

5. Discussion and Conclusion

We introduced an online method for anomaly detection in GNSS traces, grounded in the testing-by-betting framework. The method is computationally efficient and well-suited for real-time, on-vehicle deployment without the complexity of operating ML models. When running on board, it requires shuffling a small window of lat-long pairs (with linear time complexity) and evaluating a histogram-based PDF (in logarithmic time). Memory requirements are also modest: the algorithm needs to store only 90 coordinate pairs and a 360-bin reference histogram during operation.

The use of a sliding window is a simple and effective way to handle temporal dependencies in turning angles. However, it also introduces a trade-off: a window of size K leads to a detection delay of at least K steps—about three minutes in our evaluation. While this delay is acceptable for our current use case, other contexts may benefit from estimating conditional likelihoods to directly model dependencies. We leave this direction for future work.

The empirical evaluation shows that our method maintains its theoretical false positive rate bound with respect to out-of-distribution turning patterns. However, when detecting potentially reckless rides, human reviewers often rely on contextual information—such as road layout and location—that is not available to the algorithm. This leads to deviations from the expected false positive rate, which is challenging to control *a priori*. Nonetheless, the observed detection performance remains operationally acceptable. In practice, the detection output can be combined with signals from other sources, such as accelerometer and gyroscope data, to support decision-making in an automated system.

In terms of recall, the method detects approximately 60% of reckless rides. This may be improved by refining the scoring function used in the budget update. In its current form, the function compares the reference PDF to a uniform distribution, which is likely to represent only the most extreme deviations in turning behavior, missing more subtle patterns.

Overall, the method provides a lightweight anomaly detection mechanism with formal guarantees on false positive rates, validated under real-world data. Future work may include incorporating temporal dependencies into reference PDF estimation, enabling the integration of signals from other systems, and improving the scoring function.

Acknowledgments

We would like to acknowledge Hongyi Liu and Piotr Gołąb from Voi Technology for their contributions to labeling GNSS traces and to discussions on micromobility safety and anomaly detection. Further, we would like to acknowledge Rahman Amandius from Voi Technology and Erik Jenelius from the Division of Transport planning at KTH for their valuable feedback and suggestions on the manuscript. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

Laura Beggel, Michael Pfeiffer, and Bernd Bischl. Robust anomaly detection in images using adversarial autoencoders. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part I*, pages 206–222. Springer, 2020.

Giovanni Cherubin, Adrian Baldwin, and Jonathan Griffin. Exchangeability martingales for selecting features in anomaly detection. In *Conformal and Probabilistic Prediction and Applications*, pages 157–170. PMLR, 2018.

J. L. Doob. *Stochastic Processes*. Wiley, New York, 1953.

Fluctuo. European shared mobility index, 2025. URL https://european-index.fluctuo.com/. Accessed: 2025-04-25.

Indra Hansson and Julia Congreve Lifh. Anomaly detection in riding behaviours: Using unsupervised machine learning methods on time series data from micromobility services, 2022.

Shen-Shyang Ho, Matthew Schofield, Bo Sun, Jason Snouffer, and Jean Kirschner. A martingale-based approach for flight behavior anomaly detection. In *2019 20th IEEE International Conference on Mobile Data Management (MDM)*, pages 43–52. IEEE, 2019.

Rikard Laxhammar and Göran Falkman. Inductive conformal anomaly detection for sequential detection of anomalous sub-trajectories. *Annals of Mathematics and Artificial Intelligence*, 74:67–94, 2015.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pages 413–422. IEEE, 2008.

Rahul Rajendra Pai and Marco Dozza. Understanding factors influencing e-scooterist crash risk: A naturalistic study of rental e-scooters in an urban area. *Accident Analysis & Prevention*, 209:107839, 2025. ISSN 0001-4575. doi: https://doi.org/10.1016/j.aap.2024.107839. URL https://www.sciencedirect.com/science/article/pii/S0001457524003841.

Glenn Shafer. Testing by betting: A strategy for statistical and scientific communication. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 184(2):407–431, 05 2021.

Glenn Shafer and Vladimir Vovk. *Game-theoretic foundations for probability and finance*. John Wiley & Sons, 2019.

H. Stigson, I. Malakuti, and M. Klingegård. Electric scooters accidents: Analyses of two swedish accident data sets. *Accident Analysis & Prevention*, 163:106466, 2021. ISSN 0001-4575. doi: https://doi.org/10.1016/j.aap.2021.106466. URL https://www.sciencedirect.com/science/article/pii/S0001457521004978.

Jean Ville. *Etude critique de la notion de collectif*, volume 3. Gauthier-Villars Paris, 1939.

Thaddeus Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey review*, 23(176):88–93, 1975.

Vladimir Vovk, Ilia Nouretdinov, and Alexander Gammerman. Testing exchangeability online. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 768–775, 2003.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

Shumayla Yaqoob, Salvatore Cafiso, Giacomo Morabito, and Giuseppina Pappalardo. Deep transfer learning-based anomaly detection for cycling safety. *Journal of safety research*, 87:122–131, 2023.