

An Empirical Study of Machine Learning Robustness for Parking Occupancy Prediction Under Missing Data

Nikhil Saroj¹, Vidhi Bagde², AmitKumar Mehta³ and Omkar Sherkhane⁴

^{1,2,3} Student, M.Sc. Data Analytics, Pillai College of Arts, Commerce and Science, Maharashtra, India

⁴ Assistant Professor, Pillai College of Arts, Commerce and Science, Maharashtra, India

Abstract—Parking occupancy forecasting is commonly treated as a model-selection problem, but the quality and temporal structure of sensor observations can also affect forecasting performance. Parking sensors may produce missing observations because of communication failures, maintenance, equipment faults, or interruptions in data collection. When lag and rolling features are derived from occupancy history, missing observations can alter the local temporal context available to the predictor. This study investigates this effect through a controlled experiment using a cleaned Melbourne on-street parking occupancy series containing 14,062 observations at 15-minute intervals. The next 15-minute occupancy rate is forecast using four lag features, two rolling statistics, and calendar features constructed without future information. Missing observations are introduced into the raw occupancy series at six rates (5%, 10%, 20%, 30%, 40%, and 60%) under random and consecutive missingness patterns. Three reconstruction methods—Forward Fill, Linear Interpolation, and Mean Imputation—are evaluated with five prediction approaches: Persistence, Linear Regression, Random Forest, Gradient Boosting, and XGBoost. Across 180 controlled conditions, mean R^2 decreases from 0.9533 at 5% missingness to 0.3875 at 60%, while MAE increases from 0.0057 to 0.0233. Consecutive gaps are more damaging than random gaps, with mean R^2 of 0.6486 versus 0.8591. Linear Interpolation and Random Forest achieve the highest study-defined robustness scores among reconstruction and prediction methods, respectively. These findings demonstrate that parking forecasting systems should monitor both missing-data volume and temporal structure.

Index Terms—imputation, machine learning, missing data, parking occupancy prediction, time-series forecasting

I. INTRODUCTION

Cities are using parking information not only to show current availability, but also to support guidance systems, operational planning, and the management of limited curbside capacity. As these applications have expanded, parking prediction research has moved from conventional time-series models toward machine learning, deep learning, and models that combine temporal, spatial, and contextual information [1], [2]. The forecasting literature, however, often assumes that the recent observation history is complete. Real parking systems do not offer that guarantee. Sensors can stop reporting for short periods or for much longer stretches, and the effect is especially important when the forecasting pipeline derives predictors from recent occupancy values. A missing observation can reappear as a missing lag or change a rolling statistic several steps later. In that sense, missingness changes the information seen by the model itself, not merely the cleanliness of the source file.

This study focuses on that practical issue. It does not propose a new prediction architecture. Instead, it takes several established forecasting approaches and examines how their performance changes as the occupancy history is progressively damaged. The experiment also separates two conditions that may have the same percentage of missing observations but a very different temporal structure: isolated random gaps and consecutive gaps.

A. Research Problem

The research problem is the loss of short-term parking occupancy prediction quality when the observed occupancy series contains missing values. Four factors are examined together: the amount of missing data, the temporal distribution of missing observations, the

reconstruction method, and the prediction model used after reconstruction.

B. Research Gap

Parking prediction and missing-data handling are both established research areas, but they are often treated as separate topics. Parking studies tend to emphasize forecasting accuracy and model design, whereas missing-data studies focus on reconstruction assumptions and downstream effects in other domains. Reviews of missing data note that the consequences depend on the missingness mechanism, the amount of missingness, and the structure of the underlying time series [3]–[5].

The gap addressed here is narrower and deliberately practical. There are fewer studies that place a parking occupancy forecasting pipeline under a controlled combination of missingness rate, random versus consecutive gaps, simple reconstruction methods, and several prediction models. The contribution is therefore empirical: a reproducible stress test of an established forecasting pipeline rather than a claim of algorithmic novelty.

C. Research Questions

The study addresses five research questions (RQs):

- RQ1: How does increasing missingness affect predictive performance?
- RQ2: How do random and consecutive missing observations differ?
- RQ3: Which missing-data handling strategy preserves predictive performance most consistently?
- RQ4: How do the evaluated prediction models differ in robustness to missing observations?
- RQ5: At what missingness level does the prediction pipeline cross a study-defined practical performance threshold?

D. Research Hypotheses

Five hypotheses (H1–H5) correspond one-to-one to the research questions:

- H1: Increasing the proportion of missing observations will reduce prediction performance.
- H2: Consecutive missing observations will cause greater degradation than random missing observations at the same missing-data rate.

H3: Temporal reconstruction methods will preserve predictive performance better than global mean replacement.

H4: The prediction models will differ in how well they tolerate missing observations.

H5: A practical performance threshold will be crossed before the highest missingness level is reached.

E. Contributions

- A controlled experiment for measuring the downstream effect of missing parking occupancy observations.
- A comparison of random and consecutive missingness across six missingness rates.
- A direct comparison of Forward Fill, Linear Interpolation, and Mean Imputation.
- A robustness comparison across Persistence, Linear Regression, Random Forest, Gradient Boosting, and XGBoost (extreme gradient boosting).
- A study-defined practical threshold on the coefficient of determination (R^2) for translating experimental degradation into an interpretable reference point.

II. LITERATURE REVIEW

A. Parking Occupancy Prediction

Parking occupancy prediction is useful because drivers and parking operators need information about what occupancy is likely to look like in the near future, not only what is occupied at the current moment. A review of 639 publications [1] describes a field that now includes time-series methods, machine learning, deep learning, and forecasting that incorporates spatial and contextual information. A second review, covering more than one hundred studies [2], likewise reports broad use of machine learning, neural networks, hybrid models, and time-series techniques. At the application level, sensor and contextual information have been combined for parking occupancy prediction in a Web-of-Things setting [6]. These studies establish that short-term occupancy is predictable, but they also highlight an important practical dependency: the prediction quality is tied to the availability and quality of the input stream.

B. Missing Data in Machine Learning

Missing observations are a standard data-quality problem in statistics and machine learning. The

classical framework distinguishes missing completely at random, missing at random, and missing not at random, with the missingness mechanism affecting what can be learned from incomplete data [4]. A broad survey of missing-data methods in machine learning also points to operational causes such as collection failures and preprocessing errors and concludes that no single handling method is uniformly best across settings [3].

This experiment does not attempt to infer the original missingness mechanism of the Melbourne data. Instead, missing observations are imposed in a controlled manner so that the effect of missingness rate and temporal structure can be measured directly. The design should therefore be interpreted as a robustness test, not as an estimate of how often a particular real-world sensor failure occurs.

C. Missingness in Time Series

Time-series data introduce an additional complication because neighboring observations are often related. Methods that use nearby values can reconstruct short gaps more plausibly than methods that ignore time, while a continuous gap removes a whole local block of temporal information. Benchmarking work in time-series imputation shows that method performance varies with both the missingness pattern and the underlying dataset [5]. More recent work on spatio-temporal imputation also stresses the role of temporal and spatial dependence when observations are lost from sensor systems [7].

This motivates the central comparison in the present study. A random 20% loss and a consecutive 20% loss remove the same number of observations, but they do not leave the same local context around the remaining observations. The experiment treats that distinction as a factor rather than collapsing everything into a single percentage.

D. Imputation and Downstream Prediction

It is tempting to judge an imputation method only by whether the reconstructed values look plausible. For forecasting, that is not enough. A reconstruction can appear reasonable and still produce lag and rolling features that are less useful for the next-step prediction. Conversely, a simple method can preserve the information that the downstream model needs without being the most sophisticated reconstruction in a standalone imputation benchmark. Existing

benchmarking evidence shows that no single imputation method dominates across datasets and conditions [5].

The three methods in this study are intentionally simple. Forward Fill carries the last observed level forward, Linear Interpolation uses temporal neighbors around a gap, and Mean Imputation replaces the missing value with a constant occupancy level. Their simplicity helps isolate the effect of reconstruction choice without adding a large hyperparameter search.

E. Prediction Models

The model set covers several common modelling families. Persistence is a simple operational baseline. Linear Regression provides an interpretable linear benchmark. Random Forest represents a bagged decision-tree ensemble [8]. Gradient Boosting represents additive tree boosting [9]. XGBoost provides a regularized and scalable gradient-boosting implementation [10]. The aim is not to declare a universally best parking model, but to compare established approaches under the same missing-data conditions.

F. Position of the Present Study

The literature points to two ideas that are central to this experiment. First, parking occupancy is shaped by temporal and contextual patterns, so damaging the observation stream can affect the feature space as well as the raw values. Second, missing-data handling is not independent of the downstream prediction task. The same reconstruction strategy can behave differently across datasets, missingness structures, and forecasting objectives. For that reason, missingness is treated here as an experimental variable rather than a routine preprocessing detail.

III. DATASET AND METHODOLOGY

A. Dataset Source and Description

The data originate from the City of Melbourne open-data collection “On-street Car Parking Sensor Data—2020 (Jan–May).” The dataset describes in-ground parking bay sensors used across Melbourne’s central business district and covers 1 January 2020 through May 2020 [11]. The public source is much larger than the derived series used here; the present analysis uses a cleaned occupancy time series constructed from the

available records. Table I summarizes the characteristics of the series.

Table I. Dataset characteristics used in the experiment.

Characteristic	Value
Raw cleaned observations	14,062
Usable supervised observations	14,057
Raw variables	Timestamp; Occupied_Bays; Total_Bays; Occupancy_Rate
Observation interval	15 minutes
Study period	1 Jan 2020 – 26 May 2020
Total_Bays	4,935 (constant)
Target	Next 15-minute Occupancy_Rate
Raw missing values after cleaning	0
Duplicate timestamps	0

B. Data Quality and Preprocessing

- Timestamps were parsed, sorted, and checked for chronological order.
- Duplicate rows and duplicate timestamps were checked and removed where applicable; none remained in the retained series.
- The retained observations followed the intended 15-minute cadence.
- Occupied-bay values were checked for negative values and for values above capacity.
- Occupancy_Rate was checked against Occupied_Bays / Total_Bays.
- The final raw occupancy series contained no missing observations before artificial missingness was introduced.

C. Prediction Task and Feature Engineering

The forecasting target is the occupancy rate at the next 15-minute interval. For an observation at time t , the target is $y(t+1)$, while the predictors describe calendar information and occupancy history available at or before t . Table II lists the predictor groups and the target.

Table II. Predictor groups and forecasting target.

Feature group	Variables	Purpose
Temporal	Hour, DayOfWeek, IsWeekend, Month, DayOfMonth, WeekOfYear	Represent recurring calendar and time-of-day structure.
Lagged occupancy	Lag_1, Lag_2, Lag_3, Lag_4	Capture the most recent occupancy history.
Rolling statistics	Rolling_Mean_4, Rolling_Std_4	Summarise recent local level and variability using prior observations only.
Target	Target_Next_15 min	Occupancy rate at the next interval.

The order of operations is important. Rolling features are calculated from shifted occupancy values, so the current target is not included in the predictors. This prevents future information from leaking into the feature matrix and keeps the evaluation aligned with a one-step-ahead forecasting task.

D. Chronological Train, Validation, and Test Split

The data are divided chronologically into training, validation, and test partitions, as listed in Table III.

Table III. Chronological data partitions.

Partition	Observations	Start	End
Train	9,839	2020-01-01 01:00	2020-04-12 12:30
Validation	2,109	2020-04-12 12:45	2020-05-04 11:45
Test	2,109	2020-05-04 12:00	2020-05-26 11:00

The split is chronological rather than randomly shuffled. This preserves the direction of time and prevents training on observations that occur later than the validation period. The test partition is protected from the missingness manipulation and is not used to choose the reported robustness configuration.

E. Controlled Experimental Workflow

The central design choice is to introduce missingness at the raw Occupancy_Rate level and only then rebuild

the lagged and rolling features. The model therefore experiences the same information loss that a forecasting pipeline would encounter when its input stream is incomplete. Corrupting the engineered features independently would represent a different and less realistic experiment. The workflow is shown in Fig. 1.

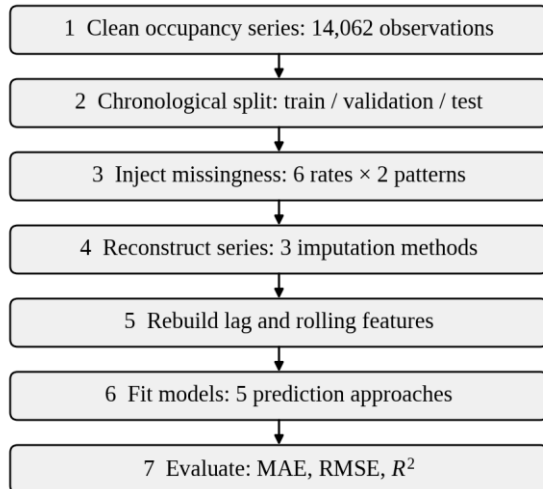


Fig. 1. Controlled experimental workflow used in the study.

F. Missingness Design

The missingness factors and their experimental boundaries are given in Table IV.

Table IV. Missingness factors and experimental boundaries.

Factor	Levels / definition
Missingness rate	5%, 10%, 20%, 30%, 40%, 60%
Random pattern	Missing observations are distributed randomly within the experimental region.
Consecutive pattern	Contiguous temporal blocks; blocks are configured around 96 observations (24 hours) where required, with adjustment to reach the target count.
Experimental region	Training + validation observations; the test partition is protected.
Random seed	42

The 60% condition is intentionally an extreme stress test. It is not treated as a normal operating assumption; its purpose is to show how the forecasting pipeline

behaves as the observation stream becomes severely degraded.

G. Missing-Data Handling

- Forward Fill: replace a missing observation with the most recent available value.
- Linear Interpolation: estimate a missing observation from neighboring temporal observations.
- Mean Imputation: replace a missing observation with a mean occupancy value.

After reconstruction, the lag and rolling predictors are rebuilt from the handled occupancy series. This sequence avoids the methodological mistake of generating temporal features from the clean series and then corrupting those features separately.

H. Prediction Models

The five prediction approaches are listed in Table V.

Table V. Prediction models used in the experiment.

Model	Role in the experiment
Persistence	Simple baseline using the most recent observed occupancy.
Linear Regression	Interpretable linear benchmark using temporal and occupancy-history predictors.
Random Forest	Bagged decision-tree ensemble.
Gradient Boosting	Additive tree-boosting model.
XGBoost	Regularized gradient-boosting implementation.

I. Model Configuration

The final experiment uses the following model configurations. Random Forest uses 300 trees with square-root feature selection and seed 42. Gradient Boosting uses 200 estimators, learning rate 0.05, depth 3, and minimum leaf size 10. XGBoost uses 300 estimators, learning rate 0.03, depth 3, minimum child weight 10, row subsampling 0.8, and column subsampling 0.8, with seed 42. Linear Regression and Persistence require no additional stochastic tuning. Only one random seed was used for the final experiment. The reported means therefore describe the factorial conditions under seed 42; they are not estimates of repeated-sampling uncertainty.

IV. RESULTS

J. Evaluation Metrics

Three regression metrics are used. Mean absolute error (MAE) is the average absolute difference between the observed and predicted occupancy rate. Root mean squared error (RMSE) gives greater weight to larger errors. R^2 measures the proportion of variance explained relative to a mean-prediction reference. The metrics are computed as

$$\text{MAE} = 1/n \sum_{i=1}^n |y_i - \hat{y}_i|, \quad (1)$$

$$\text{RMSE} = \sqrt{1/n \sum_{i=1}^n (y_i - \hat{y}_i)^2}, \quad (2)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}, \quad (3)$$

where n is the number of validation observations, y_i and $\hat{y}_i$ are the observed and predicted occupancy rates for observation i , and $\bar{y}$ is the mean of the observed occupancy rates. Equations (1)–(3) are evaluated separately for every condition.

Because the target is an occupancy proportion, an MAE of 0.0057 corresponds to about 0.57 percentage points of occupancy-rate error.

K. Degradation and Robustness Analysis

Performance degradation is measured relative to each model's own clean-data validation baseline. MAE and RMSE degradation are expressed as percentage change from the clean baseline, while R^2 loss is defined as clean R^2 minus experimental R^2 .

A study-defined robustness score is calculated as the mean of three relative components: clean MAE divided by experimental MAE, clean RMSE divided by experimental RMSE, and experimental R^2 divided by clean R^2 . Higher values generally indicate better preservation of clean-data performance. This index is descriptive and was created for this experiment; it is not a standard published metric. Because R^2 can become negative under severe degradation, the robustness index can also become negative.

L. Descriptive Aggregation

The final analysis contains 180 experimental conditions: 6 missingness rates $\times$ 2 patterns $\times$ 3 reconstruction methods $\times$ 5 models. For each pattern and rate, 15 conditions are averaged. These summaries are used to describe the main experimental factors. No confidence intervals or hypothesis tests are reported because the final experiment uses one random seed.

A. Clean-Data Baseline

The clean validation results, given in Table VI, establish the reference point for the degradation analysis. Persistence and Linear Regression have the lowest MAE at the reported precision, while Linear Regression produces the lowest RMSE and the highest R^2 . Random Forest has the weakest clean validation performance of the five approaches.

Table VI. Clean-data validation performance. MAE and RMSE are in occupancy-rate units.

Model	Validation MAE	Validation RMSE	Validation R^2
Persistence	0.0021	0.0077	0.9804
Linear Regression	0.0021	0.0076	0.9810
Gradient Boosting	0.0043	0.0080	0.9788
XGBoost	0.0059	0.0098	0.9679
Random Forest	0.0094	0.0131	0.9434

B. Overall Effect of Increasing Missingness

Table VII reports the aggregate results by missingness rate, and Fig. 2 and Fig. 3 plot the same means. The aggregate pattern is consistent across models, reconstruction methods, and missingness patterns: as missingness rises, average error increases and average R^2 declines. From 5% to 60% missingness, mean R^2 falls by 0.5658 while mean MAE increases by 0.0176. The deterioration is not perfectly linear; a noticeably larger loss appears from the 20% condition onward, with the strongest degradation occurring under consecutive gaps.

Table VII. Aggregate validation performance by missingness rate.

Missingness	Mean MAE	Mean RMSE	Mean R^2	Mean R^2 loss
5%	0.0057	0.0110	0.9533	0.0170
10%	0.0068	0.0126	0.9329	0.0374
20%	0.0108	0.0191	0.8355	0.1348
30%	0.0141	0.0234	0.7271	0.2432
40%	0.0155	0.0249	0.6871	0.2832
60%	0.0233	0.0359	0.3875	0.5828

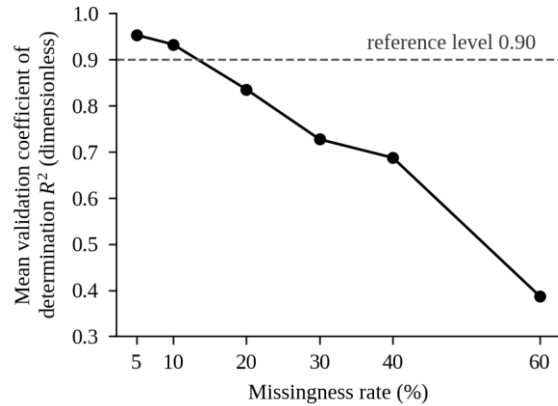


Fig. 2. Mean validation R^2 across the six missingness rates.

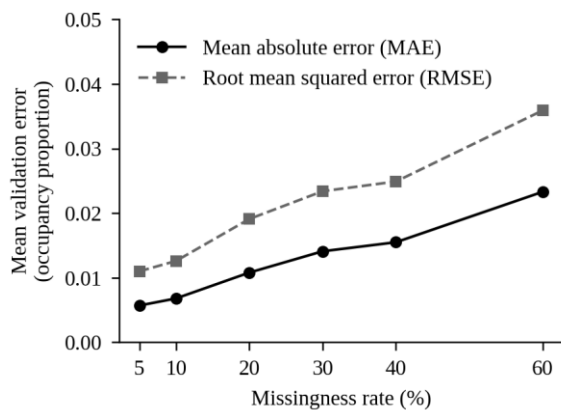


Fig. 3. Mean validation MAE and RMSE across the six missingness rates.

C. RQ1—Effect of Increasing Missingness

RQ1 is answered by the aggregate results. Higher missingness is associated with lower predictive performance across the complete set of patterns, reconstruction methods, and models. Mean R^2 decreases from 0.9533 at 5% missingness to 0.3875 at 60%, while mean MAE increases from 0.0057 to 0.0233. These results support H1 within the scope of this experiment.

D. RQ2—Random versus Consecutive Missingness

Table VIII compares the two missingness patterns. The missingness pattern has a substantial effect. Consecutive missingness produces a mean R^2 of 0.6486 compared with 0.8591 for random missingness, a difference of 0.2105 R^2 points. Mean MAE under consecutive missingness is approximately twice the random-missingness value, while mean RMSE is about 1.8 times higher.

The feature design helps explain the difference. Random gaps often leave observations immediately

before and after a missing point, so some local temporal context remains available. A long consecutive gap removes a continuous segment and deprives the model of that context for a longer period. The finding supports H2 for the conditions tested here.

Table VIII. Aggregate performance by missingness pattern.

Pattern	Mean MAE	Mean RMSE	Mean R^2	Mean robustness
Random	0.0085	0.0150	0.8591	0.8279
Consecutive	0.0169	0.0273	0.6486	0.5288

E. RQ3—Missing-Data Handling Strategy

Table IX compares the three reconstruction methods, and Fig. 4 shows their average robustness. Linear Interpolation produces the highest average robustness score (0.8288), followed by Forward Fill (0.7582). Mean Imputation is substantially weaker, with a robustness score of 0.4480. Interpolation also has the lowest aggregate MAE and RMSE and the highest aggregate R^2 among the three methods.

A plausible explanation is the local continuity of the occupancy series: nearby observations can carry useful information about the short interval between them. Forward Fill preserves the most recent level but cannot represent a gradual change across a gap, while Mean Imputation removes local temporal variation. The result supports H3, but only for this dataset and task; it does not establish that interpolation is always preferable.

Table IX. Aggregate performance by reconstruction method.

Method	Mean MAE	Mean RMSE	Mean R^2	R^2 loss	Robustness
Linear Interpolation	0.0086	0.0145	0.8912	0.0791	0.8288
Forward Fill	0.0096	0.0166	0.8757	0.0946	0.7582
Mean Imputation	0.0199	0.0323	0.4949	0.4754	0.4480

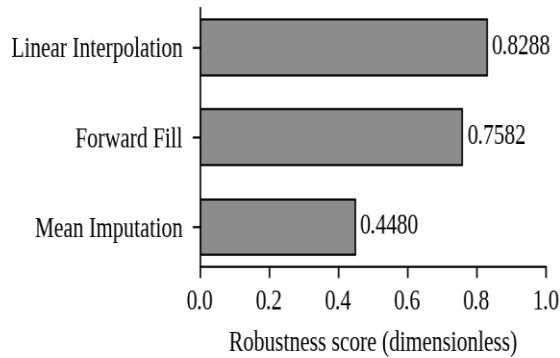


Fig. 4. Average robustness of the three missing-data handling strategies.

F. RQ4—Robustness of the Prediction Models

Table X and Fig. 5 give the corresponding comparison for the prediction models. Random Forest has the highest average study-defined robustness score at 0.8429. This should not be confused with clean-data accuracy. Linear Regression is stronger on the clean validation set by RMSE and R^2 , while Random Forest preserves more of its clean-data performance on average under the manipulated conditions. The two measures therefore answer different questions: one concerns accuracy with complete inputs, the other concerns tolerance to degraded inputs. H4 is supported within the experimental design.

Table X. Aggregate robustness of the evaluated prediction models.

Model	Mean MAE	Mean RMS E	Mean R^2	R^2 loss	Robustness
Random Forest	0.0136	0.0195	0.8327	0.1107	0.8429
XGBoost	0.0132	0.0200	0.7996	0.1683	0.7191
Gradient Boosting	0.0124	0.0186	0.8264	0.1523	0.6617
Linear Regression	0.0110	0.0191	0.8066	0.1744	0.6424
Persistence	0.0132	0.0285	0.5041	0.4762	0.5255

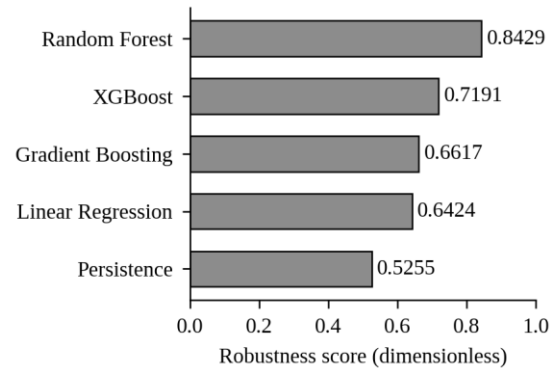


Fig. 5. Average robustness of the evaluated prediction models.

G. RQ5—Practical Missingness Threshold

Table XI reports where each missingness pattern first crosses the study-defined reference. Both patterns first fall below the study-defined $R^2 \geq 0.90$ reference at 20% missingness, but the size of the drop differs. Random missingness produces mean R^2 of 0.8943, just below the reference, whereas consecutive missingness produces 0.7767. Thus, the same percentage of missing observations does not imply the same downstream effect. H5 is supported for the experimental conditions examined.

The $R^2 \geq 0.90$ threshold is a study-defined decision rule used as a reporting reference. It is not an industry-wide standard and should not be read as a universal deployment requirement.

Table XI. First aggregate crossing of the study-defined R^2 reference.

Pattern	First rate below $R^2 \geq 0.90$	Mean R^2 at that rate
Random	20%	0.8943
Consecutive	20%	0.7767

H. Observed Extreme Conditions

Table XII lists the individual best and worst conditions observed in the factorial experiment. One individual condition produces an R^2 of 0.9910 at 60% missingness. This is retained for transparency, but it is not evidence that more missing data improve prediction. It is a single configuration inside a much larger factorial experiment, while the aggregate missingness results move in the opposite direction. At the other extreme, the Consecutive + 60% + Mean Imputation + Persistence configuration produces R^2 of

-2.1854, showing how badly the pipeline can fail when temporal context is heavily damaged and a weak reconstruction and baseline are combined.

Table XII. Individual extreme conditions observed in the factorial experiment.

Condition	Pattern, missingness, imputation, model	Value
Best MAE	Random, 60%, Linear Interpolation, Linear Regression	0.0019
Best RMSE	Random, 60%, Linear Interpolation, Linear Regression	0.0052
Best R^2	Random, 60%, Linear Interpolation, Linear Regression	0.9910
Best robustness	Random, 60%, Linear Interpolation, Linear Regression	1.1901
Worst MAE	Random, 60%, Mean, Persistence	0.0636
Worst RMSE	Consecutive, 60%, Mean, Persistence	0.0980
Worst R^2	Consecutive, 60%, Mean, Persistence	-2.1854
Worst robustness	Consecutive, 60%, Mean, Persistence	-0.7057

I. Summary of Answers to the Research Questions

Table XIII summarizes the answer to each research question.

Table XIII. Direct answers to the research questions.

RQ	Answer from the experiment
RQ1	Increasing missingness reduced performance overall; mean R^2 fell by 0.5658 from 5% to 60%.
RQ2	Consecutive missingness was more damaging than random missingness; mean R^2 was 0.2105 points lower.
RQ3	Linear Interpolation had the highest average study-defined robustness score (0.8288).
RQ4	Random Forest had the highest average study-defined robustness score (0.8429).
RQ5	Both missingness patterns first fell below the aggregate $R^2 \geq 0.90$ criterion at 20%.

V. DISCUSSION

A. Missingness as a Model-Input Problem

The main result is easier to interpret when missingness is viewed as a change in the information available to the model. The predictors are partly constructed from recent occupancy history. When those observations are missing, the model is not simply receiving one blank field; it is working with a less complete description of the recent state of the parking system. As missingness increases, that loss of information is reflected in all three evaluation metrics.

B. Why Consecutive Gaps Matter

The difference between random and consecutive missingness is one of the clearest findings in the experiment. A random 20% loss still leaves many local observations in place, whereas a consecutive 20% loss can remove an entire temporal window. The mean R^2 difference of 0.2105 points, together with the roughly twofold difference in MAE, shows that the temporal arrangement of missing observations carries information that a single missingness percentage cannot capture.

This has a practical implication for sensor monitoring. Two sites could report the same percentage of unavailable observations while exposing a forecasting model to very different conditions. A stream with many isolated one-interval gaps is not equivalent to a stream with one long outage, even before the model is retrained.

C. Why Linear Interpolation Performs Well in This Experiment

Linear Interpolation has the strongest average downstream robustness among the three reconstruction methods tested. The most direct explanation is local continuity: nearby occupancy values can provide useful information about the short interval between them. Forward Fill retains the previous level but cannot represent movement between the two sides of a gap, while Mean Imputation removes local temporal variation.

The interpretation should remain narrow. Linear Interpolation uses information from both sides of a missing interval. That is acceptable in this controlled reconstruction experiment, but it is not a strictly causal online method when the future neighbor is not yet available. A live forecasting system would need to

distinguish offline reconstruction from real-time operation.

D. Robustness versus Clean Accuracy

The clean-data baseline and the robustness analysis tell different stories. Linear Regression performs best on clean validation R^2 , while Random Forest has the highest average study-defined robustness score. This is not a contradiction; the two measures describe different properties of a forecasting system. One asks how accurate a model is when inputs are complete. The other asks how much of its clean performance remains after controlled degradation.

E. Interpreting the 20% Reference Point

The study-defined threshold is useful as a reporting device because it provides a concrete point at which aggregate performance crosses below a chosen reference. It should not be treated as a universal rule for every parking system. The more general observation is that consecutive missingness is already much more damaging than random missingness at the same 20% rate. The practical lesson is therefore less about the number 20 and more about the need to monitor missingness structure together with missingness volume.

F. Implications for Parking-Sensor Systems

- Data-quality dashboards should report both the percentage of missing observations and their temporal concentration.
- Short, isolated gaps may be recoverable with simple temporal interpolation when the local occupancy pattern is sufficiently smooth.
- Long consecutive gaps deserve closer operational attention because they remove a continuous block of predictive history.
- Model selection should include stress tests with degraded input, not only clean-data validation.
- Extreme missingness levels remain useful as stress tests even when they are unlikely during normal operation because they reveal failure regions.

VI. LIMITATIONS

- Only one Melbourne-derived occupancy dataset is evaluated, so the results cannot automatically be generalized to other cities, sensor technologies, or parking environments.

- The final experiment uses one random seed (42). Repeated-seed variability and confidence intervals are therefore not estimated.
- The robustness score is a descriptive index created for this study and is not a standard published metric.
- The $R^2 \geq 0.90$ threshold is study-defined and should not be treated as a universal deployment requirement.
- Only three simple reconstruction strategies and five prediction approaches are tested.
- Artificially injected missingness does not reproduce every real sensor-failure mechanism. Real failures may be sensor-specific, spatially correlated, demand-dependent, or otherwise non-random.
- Linear Interpolation can use observations on both sides of a gap, so it should not be interpreted as a strictly causal online method when future observations are unavailable.
- The study evaluates downstream prediction robustness rather than standalone imputation accuracy.

VII. CONCLUSION

This study treated missing sensor observations as an experimental factor in short-term parking occupancy forecasting, rather than as a routine preprocessing detail. Across 180 controlled conditions built on a cleaned Melbourne series, forecasting quality declined steadily as observations were removed, and the way the gaps were arranged in time mattered as much as how many there were.

The findings bear directly on the operation of sensor-based parking systems. A data-quality report that gives only the percentage of missing observations can hide the difference between many isolated gaps and a few long outages, so reporting both would give a more faithful picture of forecast reliability. Simple temporal reconstruction was enough to keep performance stable under moderate, scattered loss in this dataset, whereas replacement by a constant level was not, and candidate models are better judged on degraded inputs as well as on clean validation data.

The contribution is empirical rather than algorithmic: a reproducible stress test that can be reapplied to other cities, sensor types, and forecasting horizons. Directions for extending the work are listed below.

A. Future Work

- Replicate the experiment on additional cities and parking datasets.
- Repeat all conditions across multiple random seeds and report uncertainty intervals.
- Model more realistic sensor-failure mechanisms, including spatially correlated and demand-dependent outages.
- Compare additional temporal reconstruction methods and models designed to work with missingness directly.
- Examine whether practical thresholds change with forecasting horizon and parking context.
- Evaluate strictly causal online reconstruction methods where future observations are unavailable.

APPENDIX

A. Experimental Configuration

The final experimental configuration is listed in Table XIV.

Table XIV. Final experimental configuration.

Parameter	Final setting
Dataset	Melbourne on-street parking occupancy
Raw observations	14,062
Sampling	15 minutes
Prediction horizon	Next 15 minutes
Missingness rates	5%, 10%, 20%, 30%, 40%, 60%
Patterns	Random, Consecutive
Imputation	Forward Fill, Linear Interpolation, Mean
Models	Persistence, Linear Regression, Random Forest, Gradient Boosting, XGBoost
Seed	42
Total conditions	180
Evaluation	MAE, RMSE, R^2
Test partition	Protected from missingness manipulation

B. Reproducibility and Supporting Files

The experiment used the following data and output files:

- melbourne_parking_occupancy_15min_final.csv: cleaned raw occupancy dataset.
- melbourne_parking_ml_dataset_with_splits.csv: supervised dataset with chronological partitions.
- stage20_final_random_consecutive_no_knn.csv: final 180-condition missingness experiment.
- stage21_final_combined_analysis.csv: combined robustness analysis.
- stage22_final_results_tables.xlsx: result tables used for reporting.
- stage23_paper_ready_numbers.csv: paper-ready numerical summary.
- stage23_rq_evidence.csv: research-question evidence.
- stage23_final_evidence_summary.csv: final evidence summary.

C. Interpretation Guide

- Occupancy-rate MAE values are proportions. Multiply by 100 to express them as percentage points.
- R^2 values close to 1 indicate strong explained variance; a negative R^2 indicates performance worse than the mean-prediction reference.
- The robustness score is relative to each model's own clean baseline and should not be treated as a universal accuracy measure.
- Aggregate missingness-rate results are the appropriate evidence for RQ1 and RQ5; individual experimental conditions can behave differently.
- The individual 60% condition with $R^2 = 0.9910$ is retained for transparency but is not interpreted as evidence that high missingness improves prediction.

ACKNOWLEDGMENT

The authors thank Omkar Sherkhane for guidance throughout this work.

REFERENCES

- [1] C. Ma, X. Huang, and J. Li, "A review of research on urban parking prediction," *J. Traffic Transp. Eng. (Engl. Ed.)*, vol. 11, no. 4, pp. 700–720, Aug. 2024, doi: 10.1016/j.jtte.2023.11.004.
- [2] S. S. Channamallu, S. Kermanshachi, J. M. Rosenberger, and A. Pamidimukkala, "Parking occupancy prediction and analysis—A

- comprehensive study,” *Transp. Res. Procedia*, vol. 73, pp. 297–304, 2023, doi: 10.1016/j.trpro.2023.11.921.
- [3] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, “A survey on missing data in machine learning,” *J. Big Data*, vol. 8, Art. no. 140, 2021, doi: 10.1186/s40537-021-00516-9.
- [4] R. J. Little, “Missing data assumptions,” *Annu. Rev. Stat. Appl.*, vol. 8, pp. 89–107, 2021, doi: 10.1146/annurev-statistics-040720-031104.
- [5] M. Kazijevs and M. D. Samad, “Deep imputation of missing values in time series health data: A review with benchmarking,” *J. Biomed. Inform.*, vol. 144, Art. no. 104440, 2023, doi: 10.1016/j.jbi.2023.104440.
- [6] J. C. Provoost, A. Kamilaris, L. J. J. Wismans, S. J. van der Drift, and M. van Keulen, “Predicting parking occupancy via machine learning in the web of things,” *Internet Things*, vol. 12, Art. no. 100301, 2020, doi: 10.1016/j.iot.2020.100301.
- [7] S. Zahmatkesh and P. Zech, “Spatio-temporal missing data imputation: A systematic literature review with a focus on statistical and machine learning-based approaches,” *ACM Comput. Surv.*, vol. 58, no. 10, Art. no. 267, Apr. 2026, doi: 10.1145/3797903.
- [8] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [9] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.
- [10] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.
- [11] City of Melbourne, “On-street car parking sensor data—2020 (Jan–May),” City of Melbourne Open Data Portal, 2020. [Online]. Available: <https://data.melbourne.vic.gov.au/explore/dataset/on-street-car-parking-sensor-data-2020-jan-may/>