More Is More: Leveraging Missing Data in Commercial Real Estate with Machine Learning

Kahshin Leow and Thies Lindenthal [*]

Nov 27, 2025

Abstract

Missing data are pervasive in commercial real estate research, yet common practice remains to discard incomplete observations or fill gaps with crude imputation rules. We show that doing so can meaningfully distort inference and reduce predictive accuracy. Using detailed asset-level data from the NCREIF Property Index, we document substantial and systematic missingness in key variables, suggesting that data are unlikely to be missing at random. We then demonstrate that modern machine-learning methods, specifically the sparsity-aware XGBoost algorithm, can exploit incomplete observations without requiring imputation, yielding markedly higher out-of-sample predictive performance than models restricted to complete cases. Moreover, we find that incorporating incomplete data can change the apparent marginal effects and relative importance of standard covariates, implying that conclusions drawn from 'clean' subsamples may be misleading. Our results highlight that, in commercial real estate applications, more data—even if partially missing—can be more informative than smaller, perfectly complete samples.

1 Introduction

Missing data are a common problem for researchers in social sciences (Little and Rubin, 2019; Rubin, 1976). It is particularly problematic for multivariate analyses, as observations with incomplete information are dropped. One common approach is to replace missing values with means or medians, which is acceptable if the percentage of missing data is not large. Tabachnick and Fidell (2007) suggest a five percent threshold while Peng et al. (2006) suggest mean imputation is permissible provided no more than 10–20 percent of the data is missing. However, real estate researchers are seldom able to collect well-populated data sets. For example, Cannon and Cole (2011) remove 24 percent of property transactions from the proprietary National Council of Real Estate Investment Fiduciaries (NCREIF) database in their study to assess the accuracy of commercial real estate appraisals. Deppner et al. (2023),

[*] University of Cambridge. Corresponding author: K. Leow (kahshin.leow@kings.cam.ac.uk). We are grateful to Jeffrey Fisher and the National Council of Real Estate Investment Fiduciaries (NCREIF) for granting us access to NCREIF data. Daniel Ruf and Kelvin Wong are thanked for their excellent feedback and constructive criticism.

doing a similar study using a larger NCREIF database 12 years later, exclude 45 percent of the property transactions. Using imputations in such cases would be at best naive or at worst irresponsible, which is perhaps why Cannon and Cole (2011) and Deppner et al. (2023) prefer to drop incomplete observations from their studies.

The potentially bigger issue with missing data is that they may not be missing at random (Heckman, 1976, 1979), leading us to draw the wrong conclusions if transactions are dropped from our studies. In the case of NCREIF's commercial real estate database, its members are not required by regulation to submit all property-level information. They do so on a voluntary basis. The data is subsequently made available to other NCREIF members for research purposes. Therefore, properties that always report the full set of data fields may indicate that the property manager is competent. An incompetent manager, if having issues managing properties on a daily basis, is not likely to devote too much time submitting non-compulsory data reports on these properties. In addition, some managers may have an incentive to omit data fields that may be embarrassing or detrimental to future sales of their properties. For example, given the choice of reporting low capital expenditures or submitting a nil entry, a manager might prefer to report the latter.

Finally, the case for having a model that can deal with missing data is rooted in practicality. A model is as good as its usability in the real world, and the real world is full of observations that are tainted with bits and pieces of missing data, especially in real estate. A pricing model or automated valuation model (AVM) that can predict real estate values only when all data fields are fully populated is likely to be treated as a lightweight and would not be well-regarded by practitioners.

This paper investigates the ability of machine learning algorithms to help researchers overcome these issues. By definition, traditional linear methods are unable to deal with missing data. Yet, dropping large numbers of missing observations might result in biased samples that may not be truly reflective of the population. We find that machine learning models generate outperformance if they are permitted to train on longer but incomplete data. We also find that findings can greatly differ between studies that are done on small but complete datasets versus those that are done on long but incomplete datasets.

2 Literature review

Entire textbooks have been written on statistical analysis with missing data (see, e.g., Little and Rubin, 2019). In real estate research, however, the challenge of losing observations due to missing values has often been acknowledged but rarely addressed.

LeSage and Pace (2004) address the issue that hedonic models use data that only contain sold properties, while ignoring the large amount of covariance information in unsold properties simply because the dependent variable is missing. They employ a spatial estimator that predicts missing values of the dependent variable. They demonstrate improved prediction capabilities with a Monte Carlo simulation and with actual housing data. In the broader finance literature, the problem of missing data seems to be addressed somewhat, starting with Warga (1992) employing a maximum likelihood framework that accounts for missing data in the time series of U.S. government bonds. Zhou and Lai (2017) investigates the use of AdaBoost models to deal with missing data when predicting corporate bankruptcy. More recently, Freyberger et al. (2025) develop a generalized method of moments (GMM) framework to deal with missing data in cross-sectional asset pricing.

Other fields such as meteorology, healthcare, and energy have tapped more machine learning techniques to deal with the missing data problem. Wind power prediction is plagued by incomplete data collected from wind farms because of measurement error, malfunctioning sensors, and misoperation. To improve wind power prediction with missing data, Liu et al. (2018) combine an expectation-maximisation algorithm with multiple imputation approaches. In the healthcare area, Thirukumaran and Sumathi (2016) improve the prediction accuracy of diseases such as diabetes, lung cancer and breast cancer despite the presence of missing values ranging from 5 percent to 55 percent. Zaytar and El Amrani (2016) make use of a neural network algorithm called long short-term memory (LSTM) network to overcome the issue of incomplete data to improve general weather forecasts. In Park et al. (2023), a deep learning model, in particular, a multi layer perceptron (MLP), is used to estimate missing values to improve predictions of daily groundwater levels and daily soil moisture.

To our knowledge, only one study has made use of machine learning algorithms to proactively mitigate the problem of missing values in real estate: Chan et al. (2023) explore how automatic valuation models can benefit from ML techniques when missing values are present in training or prediction data. They show that gradient boosting machines with a missing value node strategy indeed predict residential property prices in Western Australia more accurately than approaches that constrain themselves to complete cases.

3 Data

The data set used for this study is provided by the National Council of Real Estate Investment Fiduciaries (NCREIF). It contains quarterly observations of all commercial properties included in the NCREIF Property Index (NPI) at the asset level, spanning 1978 through 2020. We collect information on more than 60 asset-level covariates, ranging from numerical variables such as age, net income, capital expenditure, appraisal value, loan interest, percentage leased, net rentable area, and cap rate, to dummy and categorical variables such as property type, manager group ID, MSA, and appraisal type.

We filter for all properties that have been sold, excluding partial sales and transfers of ownership. This constitutes a sample of 14,470 transactions. We lag all covariates by two calendar quarters for robustness. The descriptive statistics for the numerical variables are laid out in Table A.1 in Appendix A. The descriptive statistics for the dummy and categorical variables are presented in Table A.2 in Appendix A. There are 150 unique Manager Group IDs and 236 unique MSAs in the NCREIF dataset, with the majority of them tagged to a very small number of transactions each. For convenience, we retain the integrity of the top 50 Manager Group IDs and the top 50 MSAs based on transaction volume, and group all other Manager Group IDs and MSAs into a category named “Other”. Qualitatively, the results are unchanged whether Manager Group IDs and MSAs with low transaction volume are grouped together or not, as machine learning algorithms are able to deal with sparse and wide datasets, but OLS will have extreme difficulty in regressing an additional 286 categorical dummies when we need to use OLS for performance comparison against machine learning algorithms.

We present the frequency of missing values for numerical variables in Table 1. Columns 1 and 2 show the count and percentage of NaNs for each variable. These are clear cases of missing values as provided by NCREIF. What is less clear-cut are zero values, which are shown in Columns 3 and 4. Are these truly zero values, or do NCREIF members report zeros as a substitute for missing values? For example, can we believe that the principal repayment is zero for 77 percent of the observations when 45 percent of them have outstanding loan balances? Or can we believe that insurance expenses for 26 percent of the properties are zero? Regardless of the true answer, traditional linear methods have a tendency to overfit to the high frequency of zeros and produce inaccurate outputs, unless observations with zero values are also dropped from the data set, which in turn exacerbates the problem of missing data. Fortunately, machine learning methods that we will describe in Section 4 are suitable for dealing with such sparse data. For dummy and categorical variables, missing values are less of an issue, as seen in Table A.2 in Appendix A. Of these 12 variables, we only see the presence of missing values in two of them, namely appraisals (0.1 percent) and FundType (47.4 percent).

Table 1: Missing Numerical Variables

Variable	NaN (Count)	NaN (%)	Zero (Count)	Zero (%)	Total (Count)	Total (%)
Year	0	0.0	0	0.0	0	0.0
Age	2956	20.4	3	0.0	2959	20.4
Sq Ft	1635	11.3	0	0.0	1635	11.3
Units	10770	74.4	0	0.0	10770	74.4
Percentage Leased	2045	14.1	0	0.0	2045	14.1
Net Rentable Area	4075	28.2	0	0.0	4075	28.2
Sale Price	0	0.0	0	0.0	0	0.0
Market Value	632	4.4	0	0.0	632	4.4
Market Value_Lag1	632	4.4	0	0.0	632	4.4
Market Value_Lag2	959	6.6	0	0.0	959	6.6
Market Value per Sq Ft	1694	11.7	0	0.0	1694	11.7
Market Value per Unit	11352	78.5	0	0.0	11352	78.5
Cap Rate	6492	44.9	7	0.0	6499	44.9
NOI	632	4.4	13	0.1	645	4.5
NOI_Lag1	959	6.6	12	0.1	971	6.7
Base Rent	4412	30.5	0	0.0	4412	30.5
Contingent Income	665	4.6	13310	92.0	13975	96.6
Reimbursement Income	898	6.2	6246	43.2	7144	49.4
Other Income	1137	7.9	6089	42.1	7226	50.0
CapEx	1143	7.9	3764	26.0	4907	33.9
CapEx_Lag1	1179	8.1	3672	25.4	4851	33.5
CapEx_Lag2	1483	10.2	3665	25.3	5148	35.5
Additional Acquisition Costs	736	5.1	13308	92.0	14044	97.1
Leasing Commissions	794	5.5	10080	69.7	10874	75.2
Tenant Improvements	941	6.5	9892	68.4	10833	74.9
Building Improvements	1139	7.9	8269	57.1	9408	65.0
Building Expansion	686	4.7	13541	93.6	14227	98.3
Other CapEx	898	6.2	10488	72.5	11386	78.7

Notes: This table presents the count and percentage of NaNs and zeros for numerical variables found in the NCREIF database.

Table 1: Missing Numerical Variables (continued)

Variable	NaN (Count)	NaN (%)	Zero (Count)	Zero (%)	Total (Count)	Total (%)
Income Return	632	4.4	13	0.1	645	4.5
Capital Appreciation Return	632	4.4	4540	31.4	5172	35.8
Total Return	632	4.4	7	0.0	639	4.4
Cash Flow Return	632	4.4	9	0.1	641	4.5
Lev. Income Return	632	4.4	14	0.1	646	4.5
Lev. Appreciation Return	632	4.4	4332	29.9	4964	34.3
Lev. Total Return	632	4.4	7	0.0	639	4.4
Interest Payment	658	4.5	7758	53.6	8416	58.1
Principal Payment	661	4.6	11141	77.0	11802	81.6
Regular Principal Payment	652	4.5	11365	78.5	12017	83.0
Other Principal Payment	645	4.5	13509	93.4	14154	97.9
Loan Balance	632	4.4	7893	54.5	8525	58.9
Loan Balance_Lag1	632	4.4	7792	53.8	8424	58.2
New Financing	644	4.5	13341	92.2	13985	96.7
Admin Expense	863	6.0	4424	30.6	5287	36.6
Marketing Expense	768	5.3	7979	55.1	8747	60.4
Utility Expense	757	5.2	4564	31.5	5321	36.7
Maintenance Expense	715	4.9	3915	27.1	4630	32.0
Insurance Expense	730	5.0	3738	25.8	4468	30.8
Management Fee Expense	666	4.6	4177	28.9	4843	33.5
Tax Expense	717	5.0	3868	26.7	4585	31.7
Other Expense	1174	8.1	5622	38.9	6796	47.0
Total Expense	4128	28.5	0	0.0	4128	28.5

Notes: This table presents the count and percentage of NaNs and zeros for numerical variables found in the NCREIF database.

4 Methodology

4.1 General additive prediction error model

Throughout our analysis, we adopt a general additive prediction error model to describe the relationship between a property's transacted value and its corresponding predictors, i.e.

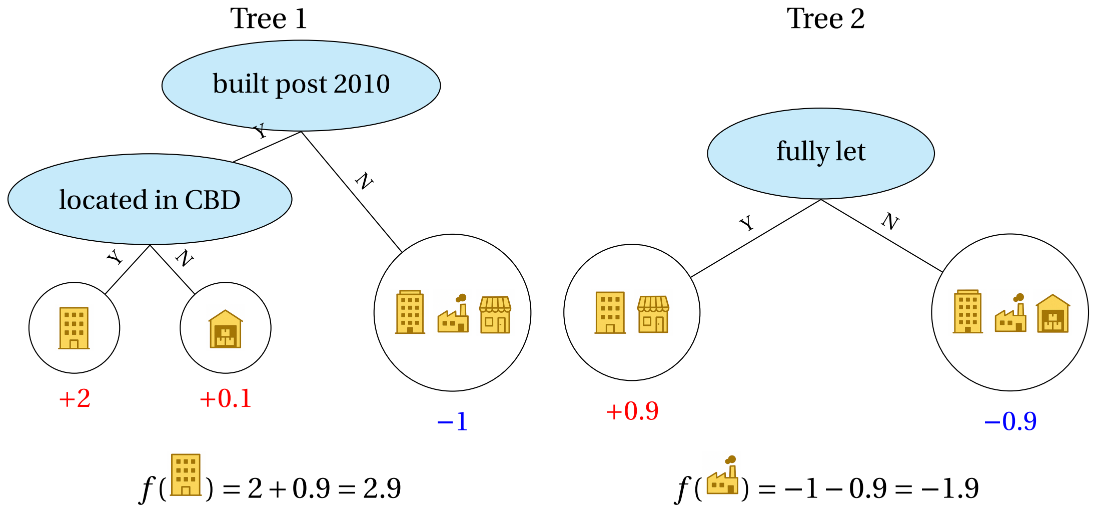
$$SalePrice_{i,t+1} = E_t[SalePrice_{i,t+1}] + \epsilon_{i,t+1}, \tag{1}$$

In addition, we further assume the conditional expectation of ith property's transacted value $SalePrice_{i,t+1}$ given the information available at period t to be a function of a set of predictors, i.e.

$$E_t[SalePrice_{i,t+1}] = g(z_{i,t}), \tag{2}$$

where $z_{i,t}$ is the baseline set of asset-level predictors, properties are indexed by $i = 1,...,N$ and quarters by $t = 1,...,T$. The functional form of $g(.)$ is left unspecified and depends on

Figure 1: Ensemble of Trees



Notes: Figure adapted from Figure 1 in Chen and Guestrin (2016). It presents a typical tree ensemble model where the final prediction for a given observation is the sum of predictions from each tree.

z only through $z_{i,t}$. This means that our prediction model does not use information from history prior to t, or from properties other than the ith property.

The vector of predictors, $z_{i,t}$, consists of the ith property's characteristics, which can be represented as:

$$z_{i,t} = \begin{pmatrix} c_{i,t} \\ d_{i,t} \\ e_{i,t} \end{pmatrix}, \tag{3}$$

where $c_{i,t}$ is a 50 x 1 vector of numerical variables, d_t is a 9 x 1 vector of categorical variables, $e_{i,t}$ is a 3 x 1 vector of dummy variables. The categorical variables, after going through the process of dummy-encoding, become a 309 x 1 vector of dummies. Hence, the total number of covariates in $z_{i,t}$ is 50 + 309 + 3 = 362.

We include time fixed effects but do not include macroeconomic predictors in our models to keep our models parsimonious and focused on teasing out the effects of missing data at the asset-level, and not at the macro-level.

4.2 Tree ensemble model

Machine learning algorithms have become important in many areas. Smart spam classifiers protect our email inboxes by learning from massive amounts of spam data and user feedback. Advertising systems optimize user clicks with the right ads. High-energy physics experiments rely on anomaly event detection systems to find events that lead to new breakthroughs. On a similar note, we shall make full use of machine learning's ability to deal with the presence of sparse data in real estate.

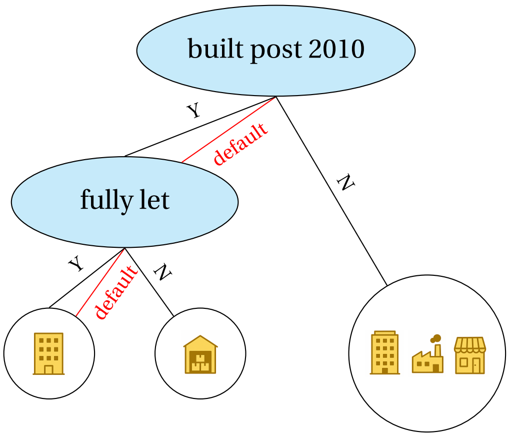
Our algorithm of choice is `XGBoost`, a well-regarded tree ensemble model developed by Chen and Guestrin (2016). A typical tree ensemble model (see, for example, Breiman 2001 and Friedman 2001) can be illustrated by Figure 1. For a given data set with n samples and m features, $\mathscr{D} = \{(\mathbf{x}_i, y_i)\}(|\mathscr{D}| = n, \mathbf{x}_i \in \mathbb{R}^m, y_i \in \mathbb{R})$, a tree ensemble model uses K additive

Figure 2: A Sparsity-Aware Tree Example

(a) Data

Property	Year of Constr.	% let
🏢	2025	NaN
🏠	NaN	75
🏢	2005	90
🏭	1989	75
🏪	1998	90

(b) Tree



Notes: Figure adapted from Figure 4 in Chen and Guestrin (2016). A tree structure with default directions. An observation will be sent into the default direction when the feature needed for the split is missing.

functions to predict the output,

$$\hat{y}_i = \phi(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i), f_k \in \mathscr{F}, \tag{4}$$

where $\mathscr{F} = \{f(\mathbf{x}) = w_{q(\mathbf{x})}\}(q : \mathbb{R}^m \to T, w \in \mathbb{R}^T)$ is the space of regression trees. The structure of each regression is represented by q, and it maps an observation to the corresponding leaf index. T is the number of leaves in the tree. Each f_k corresponds to an independent tree structure q and leaf weights w. Each regression tree contains a continuous score on each leaf, which is represented by w_i on the i-th leaf. For a given observation, the tree ensemble model will use the decision rules in all the trees (given by q) to classify it into the appropriate leaves, and calculate the final predicted value by summing up the score in the assigned leaves (given by w).

4.3 Sparsity awareness algorithm

In many real-world problems, it is quite common for the input $\mathbf{x}$ to be sparse. There are multiple possible causes for sparsity, but two typical reasons are the presence of missing values in the data and frequent zero entries in the statistics, a phenomenon that is clearly present in the commercial real estate database of NCREIF (see Table 1). In `XGBoost`, Chen and Guestrin (2016) introduces a novel sparsity-aware algorithm for parallel tree learning. They make the algorithm aware of the sparsity pattern in the data by adding a default direction in each tree node, which is illustrated in Figure 2. When a value is missing in the sparse matrix $\mathbf{x}$, the instance is sent in the default direction. There are two possible default directions in each branch and the optimal default direction is learnt from the data. The algorithm is shown in Figure B.1 in Appendix B. While there are other regression tree models such as `Spark MLLib`, `H2O`, and `R GBM` that contain sparsity-aware abilities (see Table 2), `XGBoost` is chosen as our algorithm of choice because of its speed, memory usage and accuracy over other competing algorithms.

Table 2: Comparison of Major Tree Boosting Systems

System	sparsity aware	exact greedy	approximate global	approximate local	out-of-core	parallel learning
XGBoost	yes	yes	yes	yes	yes	yes
pGBRT	no	no	no	yes	no	yes
Spark MLLib	partial	no	yes	no	no	yes
H2O	partial	no	yes	no	no	yes
scikit-learn	no	yes	no	no	no	no
R GBM	partial	yes	no	no	no	no

Notes: This table shows the comparison of the XGBoost algorithm versus other competing machine learning algorithms. Exact greedy refers to the ability to enumerate over all possible splits on all the features in the data set. Approximate global refers to the ability to propose all the candidate splits during the initial phase of tree construction, and uses the same proposals for split finding at all levels. Approximate local refers to the ability to generate a new set of candidate split points for each node as the tree is being built. The global method requires less proposal steps than the local method. Out-of-core refers to the ability to utilize disk space to handle data that does not fit into main memory. Parallel learning refers to the ability to train with multiple CPU cores.

Importantly, XGBoost does not impute missing values. Instead, during training it learns an optimal default direction for missing observations at each split ('sparsity-aware split finding'). Thus, the model uses patterns in missingness directly rather than filling in values.

4.4 Walk-forward validation versus k-fold cross-validation

To benchmark the predictive power of the models, we adopt the walk-forward validation method that Leow and Lindenthal (2025) use, rather than the k-fold cross-validation method commonly adopted by other real estate researchers in machine learning. Walk-forward analysis requires dividing our data into two disjoint periods while maintaining the temporal ordering: the training sample and the testing sample. We use the training sample to estimate the model parameters. The testing sample contains the next 12 months of data right after the training sample ends. These data, which never enter into model parameter estimation, are used to test our models' prediction performance. When one uses k-fold cross-validation on real estate data, such as Ho et al. (2021) and Deppner et al. (2023), one may inadvertently introduce time contamination into the training set. This is because k-fold cross-validation randomly splits the entire dataset into k groups of the same size, with the model training k times on $k-1$ folds and tested on the kth fold. To elaborate, each observation in the data set is assigned to an individual fold and stays in that fold for the duration of the cross-validation procedure. This means that each sample is given the opportunity to be used in the hold-out set one time and used to train the model $k-1$ times. While this technique works well for most machine learning problems, such as image recognition or anomaly event detection, it creates an unfair and unrealistic advantage when applied to time series data in real estate. For example, if sale transactions during the Great Financial Crisis of 2007–08 (GFC) are randomly assigned across 10 folds [1], a well-tuned machine learning model can easily fit to the historically low price per square footage associated with MSAs severely affected by the GFC in the *training* folds, and come up with accurate predictions for transactions occurring the GFC period in the same MSAs within the *test* folds. This problem is exacerbated if one augments asset-level variables with macroeconomic variables such as GDP, employment rate and government bond yields, which Deppner et al. (2023) did. It is not uncommon to see high R^2s nearing 99 percent in such machine learning studies, whereas the R^2s in our study range between 80–90 percent because we maintain temporal order by "walking forward".

In our study, we do not require a validation sample as we do not perform any hyperparameter optimization following Elkind et al. (2022). Default hyperparameters are used where possible. This forms the lower bound of performance for our machine learning models. Appendix C provides more information on default hyperparameters. All training is executed with open source libraries on an Apple M1 Ultra chip with a 20-core CPU and a single 48-core GPU.

4.5 Performance metrics

To measure the accuracy of our model's predictive performance with and without missing values, we calculate the out-of-sample predictive R^2, by test year according to the walk-forward procedure, and across the entire test period. This is an indicator of whether the model predicts *individual* sale prices well. However, most NCREIF members invest in a portfolio of properties, so they may be more interested in the value of the portfolio rather than in the values of individual properties in the portfolio. Therefore, a mean percentage error (MPE) metric,

$$MPE = \frac{1}{n}\sum_{i=1}^{n}\frac{SalePrice_i - PredictedPrice_i}{PredictedPrice_i}, \tag{5}$$

where positive and negative individual errors cancel out, is more informative for real estate investors. We adopt MPE as our second performance metric.

4.6 Variable importance and marginal relationships

We aim to identify NCREIF covariates that have an important influence on the cross-section of expected commercial property prices, while simultaneously controlling for other predictors in the data set. We discover influential covariates by ranking them according to the concept of variable importance, which we denote as VI_j for the jth predictor. Like Gu et al. (2020), we calculate the reduction in predictive R^2 from setting all values of predictor j within each training sample to zero, while holding the remaining predictors fixed. We average them into a single importance measure for each predictor.

As part of our analysis, we also trace out the marginal relationship between expected property values and each predictor. Despite obvious limitations, such a plot is an effective tool for visualizing the first-order impact of covariates in a machine learning model.

[1] A commonly used parameter for k-fold cross validation is k=10. See Ho et al. (2021) and Deppner et al. (2023).

Table 3: Yearly breakdown of NCREIF sale transactions, with and without missing values

Year	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*	*Year*	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*
1978	5	0	-100.0	2000	318	27	-91.5
1979	2	0	-100.0	2001	302	34	-88.7
1980	3	0	-100.0	2002	334	71	-78.7
1981	4	0	-100.0	2003	417	79	-81.1
1982	23	0	-100.0	2004	607	101	-83.4
1983	47	0	-100.0	2005	771	148	-80.8
1984	76	0	-100.0	2006	664	127	-80.9
1985	97	0	-100.0	2007	603	137	-77.3
1986	124	0	-100.0	2008	251	51	-79.7
1987	91	0	-100.0	2009	252	56	-77.8
1988	129	0	-100.0	2010	317	72	-77.3
1989	149	0	-100.0	2011	410	109	-73.4
1990	110	0	-100.0	2012	619	206	-66.7
1991	107	0	-100.0	2013	863	323	-62.6
1992	97	0	-100.0	2014	770	291	-62.2
1993	156	0	-100.0	2015	641	253	-60.5
1994	182	0	-100.0	2016	784	350	-55.4
1995	185	1	-99.5	2017	676	175	-74.1
1996	370	0	-100.0	2018	581	162	-72.1
1997	452	0	-100.0	2019	690	263	-61.9
1998	399	0	-100.0	2020	473	145	-69.3
1999	319	0	-100.0	Total	14470	3181	-78.0

Notes: This table reports the yearly breakdown of NCREIF sale transactions, with and without missing values. Columns 2 and 6 display the number of observations (i.e. sale transactions) for each year. Columns 3 and 7 display the number of observations that do not contain missing values within the 63 data fields shown in Table A.1 and Table A.2, with the exclusion of Units and Market Value per Unit. Columns 4 and 8 calculates the percentage of observations that one would have to discard if one only uses observations without missing data.

5 Empirical analysis

5.1 Full sample analysis

Table 3 displays the count of NCREIF properties that have been sold on a yearly basis, and whether they contain missing data. Appendix D shows the breakdown of missing values by property type. As we are using walk-forward validation instead of pooling all observations for a k-fold cross-validation, Table 3 is important in helping us decide when to start testing our model. Columns 2 and 6 show all properties that were sold between 1978 and 2020, whether or not they contain missing values. Columns 3 and 7 show the number of properties that do not contain missing values in any of the covariates[2] shown in Tables A.1 and A.2.

[2] For practical reasons, we exclude *Units* and *Market Value per Unit* from the empirical analysis because of their extremely high level of missing values. While XGBoost will perform better with the inclusion of these

We have a total of 14,470 properties that were sold between 1978 and 2020, and a subset of 3,181 properties that do not have missing values. This represents a 78 percent loss in the number of observations should researchers wish to model on a clean and tidy data set. For comparison, Deppner et al. (2023) start with 12,956 properties from 1997 through 2021, and after filtering, end up with a sample of 7,133 properties. This represents a data loss of 45 percent, which is lower than our study as we make full use of the range of covariates that the NCREIF database provides. Cannon and Cole (2011) start with 9,439 properties from 1982 through 2010, and after filtering, end up with a sample of 7,214 properties. This represents a data loss of 24 percent, as they use fewer covariates than Deppner et al. (2023) in their study. In Section 5.2, we shall intentionally drop some covariates to achieve a level of data loss that is comparable to Cannon and Cole (2011), but we will show that retaining more covariates, even if they are full of missing data, is still superior to having less covariates when it comes to predictions and understanding the behaviour of real estate transaction prices.

Given the dearth of complete observations from the 1980s and 1990s, we start training all models in 2000 and conduct walk-forward tests from 2001 through 2020. In Table 4, in Columns 2 and 4, we report the out-of-sample R^2s of an `XGBoost` model trained and tested on properties without missing values and with missing values, respectively. We do not display the results of simple OLS models that are trained on the same data sets. With more than 300 covariates described in Section 4, most of which are sparse data populated by zero values, the OLS model is unable to cope and generates negative out-of-sample R^2s. On the other hand, a tree-ensemble model like `XGBoost` that is sparsity-aware is able to cope and generate high positive out-of-sample predictability. The R^2_{oos} of `XGBoost` that is trained on properties with no missing values is 84.12 percent while R^2_{oos} of `XGBoost` that is trained on properties with missing values is 91.66 percent, an improvement of 7.54 percentage points. If one were to use a 10-fold cross-validation to evaluate the models instead of using walk-forward testing, the R^2s jump to 93.02 percent and 94.47 percent respectively (versus 84.12 percent and 91.66 percent for walk-forward), a possible sign of data leakage. If one were to include the time series of macroeconomic variables into the data set, the “out-of-sample” R^2s of the 10-fold cross-validation will jump to nearly 100 percent. Therefore, researchers should be careful in using k-fold cross-validations when applying data sets to machine learning models. In Table E.1 of Appendix E, we display the performance results of the same analysis using natural log of variables. The results are qualitatively unchanged from Table 4.

One might argue that it is perhaps unfair to compare the performance of Model 1 with Model 2, as the test sets are different. Model 1’s test set contains 3,153 complete observations from 2001 through 2020, while Model 2’s test set contains 11,205 NaN-filled observations from 2001 through 2020. Model 3 addresses the issue by training on incomplete NaN-filled observations across the same time period as Model 1, but conducts its test on observations with no missing values, which is exactly the same test set as Model 1. This makes for a fairer comparison with Model 1 – to see if researchers should stick to the practice of throwing away observations with incomplete information, or make use of advanced machine learning techniques to deal with missing data that is commonplace in the real world. Column 6 shows that the R^2_{oos} is 95.51 percent, a large jump of 11.39 percent over Model 1 which is trained by a data set with no missing values.

two sparse data fields in our analysis, using them will necessitate the elimination of 93 percent of the sale transactions when we create a "No NaN" data set for direct comparisons.

Table 4: Out of sample R^2 and MPE by year

	Model 1		*Model 2*		*Model 3*		*Model 4*			
Train Set	No NaNs		With NaNs		With NaNs		With NaNs			
Test Set	No NaNs		With NaNs		No NaNs		No NaNs			
Train Start	2000		2000		2000		1978		*R^2 difference*	
Test Start	2001		2001		2001		2001		*between models*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	1 vs 3	1 vs 4
2001	-1121.87	21.59%	91.40	1.53%	97.62	3.28%	96.74	0.29%	-1219.49	-1218.61
2002	84.38	3.83%	80.13	-14.86%	88.96	-0.82%	94.82	11.90%	-4.58	-10.44
2003	91.24	-0.87%	89.29	23.51%	95.32	1.59%	89.24	-2.50%	-4.08	2.00
2004	88.27	3.75%	69.98	15.41%	98.55	2.07%	98.84	0.72%	-10.28	-10.57
2005	94.37	10.89%	87.81	8.02%	87.08	14.88%	89.06	8.40%	7.29	5.31
2006	92.52	0.04%	87.12	13.35%	96.61	2.12%	95.95	0.62%	-4.09	-3.43
2007	80.86	-1.02%	76.67	-1.06%	94.90	-1.90%	95.41	-3.82%	-14.04	-14.55
2008	96.72	-7.15%	83.93	-13.00%	84.23	-7.11%	89.24	-2.83%	12.49	7.48
2009	60.16	-20.23%	78.17	-25.63%	87.53	-15.77%	86.52	4.62%	-27.37	-26.36
2010	97.25	6.10%	94.43	21.84%	96.88	9.11%	96.91	17.72%	0.37	0.34
2011	97.07	8.41%	89.80	2.40%	97.61	8.22%	98.32	12.38%	-0.54	-1.25
2012	96.38	-3.68%	92.64	-25.66%	94.04	-1.61%	94.82	-2.12%	2.34	1.56
2013	89.52	2.97%	96.55	1.27%	96.97	5.30%	97.21	8.45%	-7.45	-7.69
2014	97.27	0.89%	93.98	4.35%	98.31	3.72%	97.24	2.95%	-1.04	0.03
2015	89.89	2.25%	94.06	2.40%	93.38	1.99%	97.19	5.16%	-3.49	-7.30
2016	85.02	-2.14%	93.58	-2.79%	93.80	-3.38%	93.59	-2.07%	-8.78	-8.57
2017	97.59	-2.43%	92.00	-0.02%	96.58	-2.36%	96.28	-1.44%	1.01	1.31
2018	90.11	-0.74%	96.79	6.15%	99.17	3.27%	98.14	0.05%	-9.06	-8.03
2019	96.85	0.01%	97.47	0.09%	98.18	-1.13%	98.60	3.39%	-1.33	-1.75
2020	95.55	4.98%	94.15	-9.07%	95.54	2.42%	96.84	1.77%	0.01	-1.29
All Years	84.12	0.99%	91.66	1.31%	95.51	1.54%	96.21	2.83%	-11.39	-12.09
All Years ex ’01	91.69	0.76%	91.65	1.31%	95.49	1.52%	96.20	2.85%	-3.80	-4.51

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Perhaps, it is not necessary to artificially constrain oneself to being “fair” when employing machine learning techniques to deal with missing data. The rationale for digesting observations with missing data is that every observation is valuable, even if it is an incomplete observation. Therefore, in Model 4, we go further back to 1978 to train our model, even though the NCREIF database in the 1970s, 1980s and 1990s is full of incomplete observations. The out-of-sample performance test is then conducted Model 1’s test set (i.e. the 3,153 complete observations from 2001 through 2020), for a direct comparison to Model 1 which is typically favoured by researchers. Column 8 shows that the out-of-sample predictive performance increases to 96.21 percent, marking a 12.09 percentage points jump in R^2_{oos}. Nevertheless, we note that the MPE increases from Model 1 to Model 4, meaning that on a portfolio basis, the advantages of incorporating observations with missing values may not be as good as they are made out to be. Of course, the counter-argument is that the real world is full of observations that are tainted by missing values. What good is a model if it is unable to make a prediction when an observation is missing some values, which is the main

Table 5: Out of sample R^2 and MPE by property type

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Property Type	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
Industrial	66.17	3.30%	69.16	3.73%	68.99	1.83%	62.63	2.15%
Office	83.55	1.42%	88.65	3.61%	93.08	-1.50%	93.24	-0.57%
Apartment	75.92	2.04%	87.40	1.90%	81.76	1.05%	83.15	0.27%
Retail	76.58	10.74%	89.36	4.96%	94.98	5.09%	95.84	1.55%
Hotel	81.50	-0.59%	81.65	16.79%	88.88	25.42%	78.29	26.85%
All	84.12	0.99%	91.66	1.31%	95.51	1.54%	96.21	2.83%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models, by property type. R^2s are expressed as a percentage.

issue with Model 1, a model that is commonly preferred by researchers in the past? In addition, the increasing MPE from Model 1 to Model 4 is not observed when we start expanding our analysis by property type.

Table 5 summarises the out-of-sample performance by the property types[3]. Detailed performance reports are found in Appendix F. Looking at Table 5, the R^2s in Column 6 are consistently higher than the R^2s in Column 2 across all property types, with the biggest jump of 18.4 percent in the retail sector. The MPE also improves for all sectors except for Hotel. This suggests that a model trained by incomplete data is superior to a nice but smaller data set of complete observations.

At the risk of repeating the obvious, if one were to use k-fold cross-validation for performance assessment, we will see the R^2s jumping across the board, a possible sign of data leakage. For example, Model 1's R^2s for Office and Apartment are 95.91 percent and 92.34 percent respectively under 10-fold cross-validation, as opposed to 83.55 percent and 75.92 percent respectively under walk-forward validation.

5.2 Reducing data loss by removing covariates

Using all the variables that the NCREIF database provides might lend itself to criticism that the data loss is too high if one were to strictly exclude any observation that has at least one missing value. Indeed, in Table 1, we observe a data loss of 78 percent when we exclude observations with missing values. In contrast, Deppner et al. (2023) and Cannon and Cole (2011) experience data loss of 45 percent and 24 percent, respectively, by choosing to employ less covariates in their analysis. From a machine learning perspective, any data is good data as long as one knows how to make good use of it, but as a robustness check, we shall remove the top 10 data fields in terms of missing data from the training data set[4]. Table 6 shows

[3] Appendix F displays the performance metrics (R^2 and MPE) by year and by property type. The property type *Hotel* has a shorter train and test period because it has more observations with missing values in the earlier years than other property types.

[4] These data fields are *Cap Rate,Base Rent,Total Expense, Net Rentable Area, Age, Percentage Leased, Market Value per Square Feet, Square Feet, CapEx Lag 1, CapEx Lag 2.*

Table 6: Properties with missing values (Full and Reduced data sets)

Property Type	*Data Set*	*With NaNs*	*No NaNs*	*Loss (%)*
All	Full	14470	3181	-78.0
All	Reduced	14470	10605	-26.7
Industrial	Full	4964	931	-81.2
Industrial	Reduced	4964	3567	-28.1
Office	Full	3738	803	-78.5
Office	Reduced	3738	2620	-29.9
Apartment	Full	3402	952	-72.0
Apartment	Reduced	3402	2696	-20.8
Retail	Full	2077	377	-81.8
Retail	Reduced	2077	1474	-29.0
Hotel	Full	289	118	-59.2
Hotel	Reduced	289	248	-14.2

Notes: This table reports the number of NCREIF properties that have been sold, with and without missing data. Full refers to the data set that utilises the 63 data fields provided by NCREIF, while Reduced removes the top 10 data fields that contains most missing data, namely *Cap Rate,Base Rent,Total Expense, Net Rentable Area, Age, Percentage Leased, Market Value per Square Feet, Square Feet, CapEx Lag 1, CapEx Lag 2.*

a much smaller data loss of 26.7 percent for across all NCREIF properties, with data losses ranging from 14.2 percent to 29.9 percent when split across property types, bringing data losses in line with what Cannon and Cole (2011) and Deppner et al. (2023) experience in their studies.

Table 7 shows the out-of-sample performance on the narrower data set, for all properties and across different property types. Appendix G contains the detailed performance reports, broken down by year and by property type. True enough, if one were to look at Model 1, which is trained only on observations with no missing values, the performance improves, with R^2_{oos} increasing from 84.12 percent to 92.49 percent. This is to be expected, as the data loss drops from 78.0 percent to 26.7 percent. In Model 3, we see an improvement in R^2_{oos} to 94.18 percent, once the machine learning model is permitted to train on observations with missing values in the reduced data set, once again proving that it is important not to discard such observations. Such improvements are also seen across all property types (except *Industrial*). Perhaps, what is more interesting is that while reducing the number of data fields improves performance for Model 1, it does not translate to an R^2_{oos} improvement for Model 3 or Model 4. Again, this hints that with the help of machine learning algorithms, more covariates are better, even if the “more” is plagued with missing values.

Table 7: Out of sample R^2 and MPE (Full and Reduced Data Sets)

		Model 1		*Model 2*		*Model 3*		*Model 4*	
Property Type	Data Set	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
All	Full	84.12	0.99%	91.66	1.31%	95.51	1.54%	96.21	2.83%
All	Reduced	92.49	1.55%	91.85	-0.68%	94.18	-4.26%	93.21	2.00%
Industrial	Full	66.17	3.30%	69.16	3.73%	68.99	1.83%	62.63	2.15%
Industrial	Reduced	77.28	1.18%	71.73	5.37%	74.75	1.65%	56.31	2.45%
Office	Full	83.55	1.42%	88.65	3.61%	93.08	-1.50%	93.24	-0.57%
Office	Reduced	89.39	1.41%	90.25	6.33%	92.72	2.58%	92.69	3.42%
Apartment	Full	75.92	2.04%	87.40	1.90%	81.76	1.05%	83.15	0.27%
Apartment	Reduced	88.44	0.66%	88.18	1.48%	89.05	1.28%	89.33	1.21%
Retail	Full	76.58	10.74%	89.36	4.96%	94.98	5.09%	95.84	1.55%
Retail	Reduced	83.19	3.28%	89.22	3.11%	93.31	2.88%	93.70	3.94%
Hotel	Full	81.50	-0.59%	81.65	16.79%	88.88	25.42%	78.29	26.85%
Hotel	Reduced	82.97	8.79%	85.92	21.38%	83.30	27.42%	89.53	19.18%

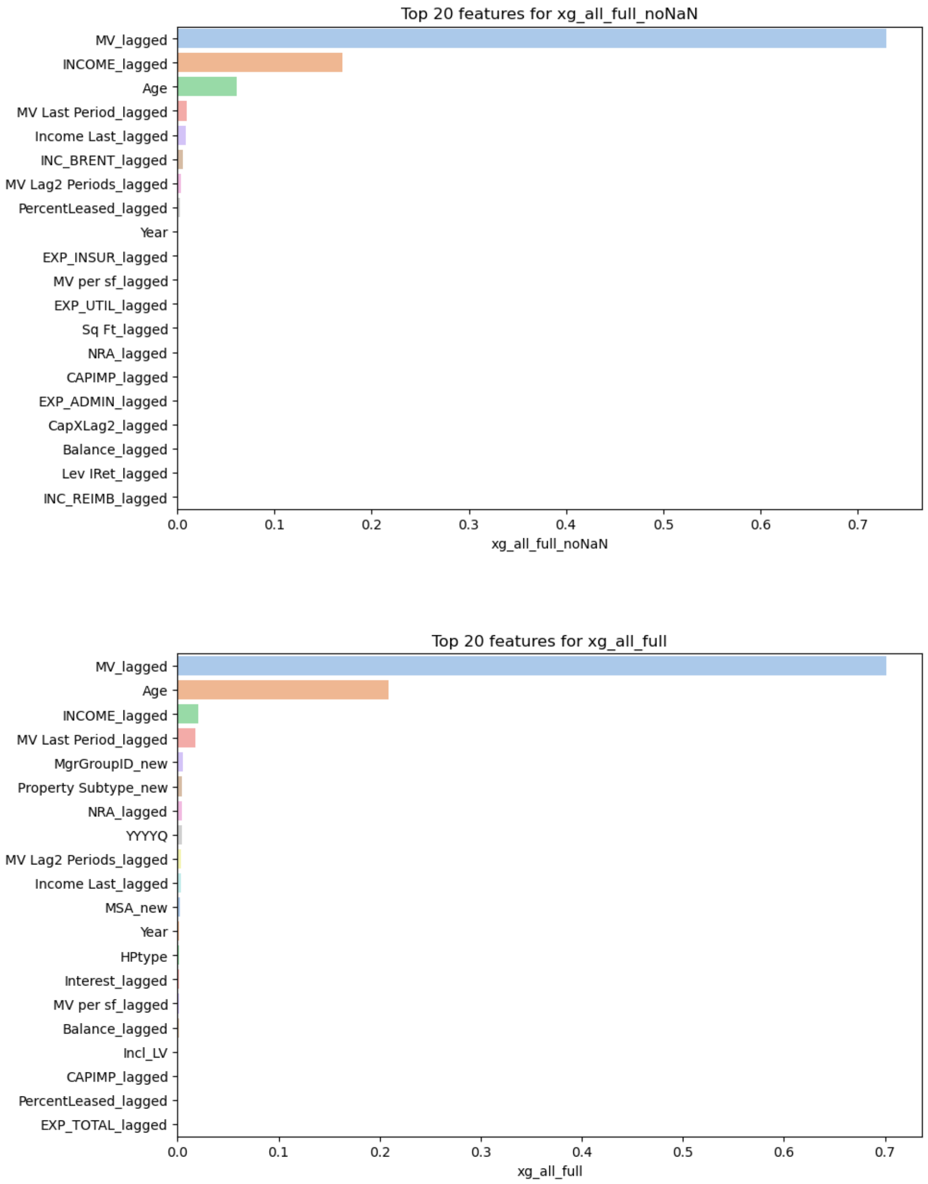
Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage. Full refers to the data set that utilises the 63 data fields provided by NCREIF, while Reduced removes the top 10 data fields that contains most missing data, namely *Cap Rate,Base Rent,Total Expense, Net Rentable Area, Age, Percentage Leased, Market Value per Square Feet, Square Feet, CapEx Lag 1, CapEx Lag 2.*

5.3 Which covariates matter?

We now investigate the relative importance of individual covariates in each model using the variable importance measure described in Section 4.6. To begin, for each model, we calculate the reduction in R^2 from setting all the values of a given predictor to zero within each training sample, and average them into a single importance measure for each predictor. Figure 3 reports the resultant importance of the top-20 asset-level characteristics for a model that is trained on a data set without missing values and with missing values, respectively. Variable importance within each model is normalized to sum to one, allowing for the interpretation of relative importance for each model. Figure 4 reports overall rankings of the 61 asset-level characteristics for both models. We rank the importance of each variable for each model, then sum their ranks. Variables are ordered so that the highest summed ranks are on the top, and the lowest ranking variables are at the bottom. The color gradient within each column shows the model-specific ranking of variables from least (white) to most important (dark blue).

Figure 3 demonstrates that market appraisal is the single most important variable of importance for both models, occupying slightly more than 70 percent weight in terms of relative importance to both models. This is perhaps not surprising, as market appraisal is an

Figure 3: Top-20 most influential variables

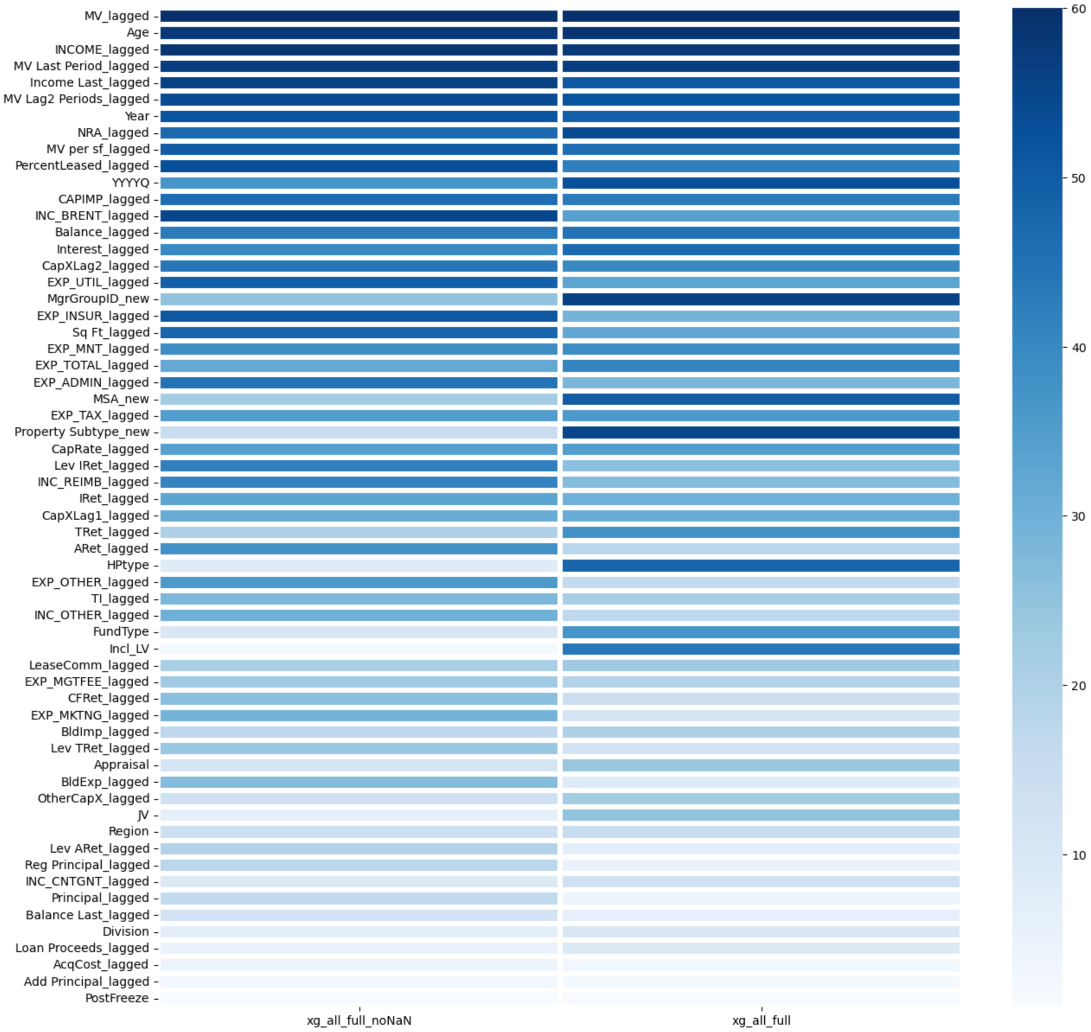


Notes: Variable importance for the top-20 most influential variables in each model. The top panel is Model 1, which is trained on observations without missing value. The bottom panel is Model 3, which is trained on observations that include missing values. Variable importance within each model is normalized to sum to one.

important industry that values more than \$20.7 trillion[5] worth of U.S. commercial real estate yearly. Perhaps what is more interesting is that *Age*, which is missing in 20.4 percent of observations (see Table 1), is considered the second most important variable for Model 3, which is trained on observations with missing values. Additionally, there are a good number of variables that are present in the top-20 ranking for Model 3 but entirely missing in Model 1, namely Manager Group ID, Property Subtype, Property Type, Interest Payment, Leverage

[5] Source: NAREIT study as of 2021:Q2

Figure 4: Heatmap of variable importance by model



Notes: Rankings of 61 asset-level variables in terms of overall model contribution. Variables are ordered based on the sum of their ranks over both models, with the most influential characteristics on the top and the least influential on the bottom. Columns correspond to the individual models (without missing values on the left column, with missing values on the right column), and the color gradient within each column indicate the most influential (dark blue) to the least influential (white).

indicator, Total Expense. Conversely, there are some variables that are present in the top-20 ranking for Model 1 but entirely missing in Model 3, namely Base Rent, Insurance Expense, Market Value per Sq Ft, Utility Expense, Administrative Expenses, Leveraged Income Return, Reimbursement Income. This shows that while both models may use the same data fields as inputs, the presence of missing values can greatly change the relative importance of variables for predicting and understanding the behavior of commercial property prices.

Figure 4 is a heat map that gives a better picture of the similarities and differences between the two models. First, both models are in general agreement on the main variables (i.e., market value and net operating income, with varying lags). Second, both models are in general agreement on the bottom few variables that are deemed unimportant (i.e., loan proceeds from new financing, acquisition cost, other principal payments). Third, the model that is trained on missing values relies more on categorical variables than the model that is trained on complete information. We see a few standout variables in Manager Group ID, MSA, Property Types and Subtypes, Fund Types, Leverage Indicator, Appraisal Type, Joint Venture Indicator. This is perhaps understandable. As the issue of missing variables are mostly limited to numerical variables (as seen in Table 1) while categorical variables are mostly complete (see Table A.2), a model that is compelled to take into account both missing and non-missing values will naturally rely more on variables that tend to have fewer missing values, i.e. categorical variables.

5.4 Marginal association between covariates and expected prices

Figure 5 traces out the model-implied marginal impact of individual asset-level variables on expected transaction prices. Despite obvious limitations, such plots are important in helping us visualize and differentiate the first-order impact of covariates when we introduce missing values into the picture. We choose six illustrative variables for Figure 5. The first four are Market Value, Age, Income and Net Rentable Area, which are variables that are commonly agreed by both Model 1 and Model 3 as important. The fifth is Insurance Expense, which is a variable that is highly ranked by Model 1 but not found in Model 3's top rankings. The sixth is Interest Payment, which is a variable that is highly ranked by Model 3 but not found in Model 1's top rankings.

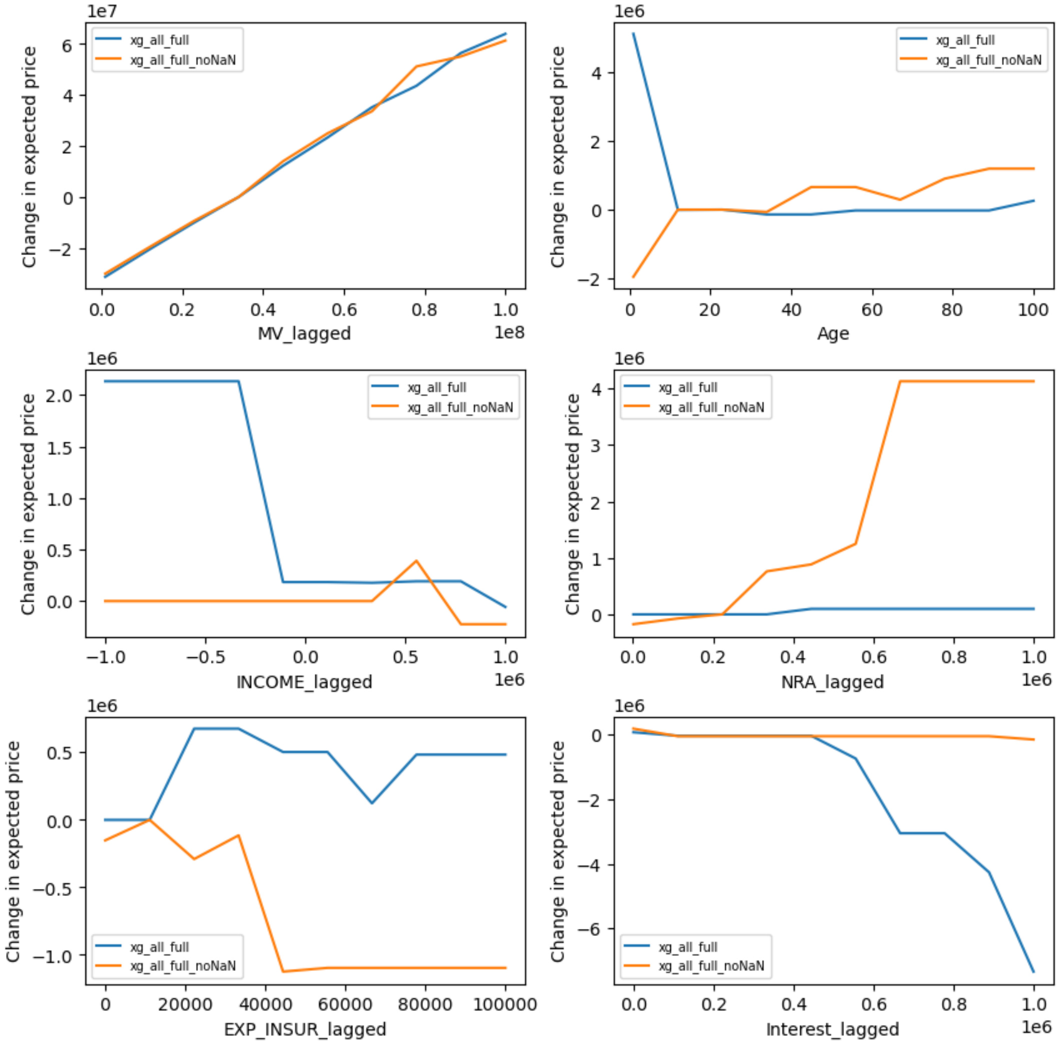
In the top left panel of Figure 5, which displays the relationship of expected prices and lagged appraised market values, we see a near-perfect linear relationship between expected price and market value. However, we start seeing an interesting deviation when we move to the top right panel, which displays expected price versus property age. For the model that is trained on complete observations (Model 1), we see a positive relationship between price and age, with the relationship being steeper for properties younger than 15 years and gentler for properties older than that. For the model that is trained on a larger set of incomplete information (Model 3), it does not give much credibility to age when a property is older than 15 years. The slope is practically flat. However, before 15 years of age, there is a sharp negative relationship between price and age, meaning that young or new commercial properties command a price premium, all else equal. This is more believable than Model 1's relationship, at least in the realm of young commercial real estate.

In the middle left panel of Figure 5, which displays the expected price versus the net operating income (NOI), we observe a deviation between the two models. When NOI turns negative, Model 3 expects a property to be more valuable, whereas Model 1 does not predict a change in expected price. A negative NOI has two possible explanations: a large drop in gross income, or a large increase in expenses. In the case of Model 3, it is possible that the negative slope is associated with a large increase in expenditures, which may imply that an owner is beautifying or window-dressing a property for sale. However, Model 1 cannot detect any implied movement in expected prices for negative NOI.

In the middle right panel of Figure 5, which displays the expected price versus the net rentable area (NRA), Model 3 flattens throughout all NRA values, while Model 1 has a positive slope for most values until it flattens above 700,000 sq ft. This implies that Model 1 relies more on NRA than Model 3 to make price predictions.

In the bottom left panel of Figure 5, we observe the marginal effect of insurance expense

Figure 5: Marginal association between expected price and asset-level features



Notes: The panels show the sensitivity of expected transaction prices (vertical axis) to the individual characteristics of properties (holding all other covariates fixed at their median values).

on the expected price. This variable is listed amongst the Top 20 for Model 1, which is trained on complete observations, but not Model 3. Again, we see a big difference in marginal associations. A higher insurance expense implies a higher expected property price for Model 3, while a higher insurance expense implies a lower property price for Model 1. It is difficult to derive an explanation for the deviations, as a higher insurance expense might imply a more valuable property as assessed by the insurance company, or it might imply a lower operating income for the owner, which may translate to lower property prices.

In the bottom right panel of Figure 5, we observe the marginal effect of interest payment on expected price. This variable is listed amongst the Top 20 for Model 3, which is trained on incomplete observations, but not Model 1. Not surprisingly, Model 1 does not detect any marginal impact on prices when interest payment changes, showing a flatline across all

Table 8: Out of sample R^2 and MPE

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Train Set	No NaNs		With NaNs		With NaNs		With NaNs	
Test Set	No NaNs		With NaNs		No NaNs		No NaNs	
Train Start	2000		2000		2000		1978	
Test Start	2001		2001		2001		2001	
Data Type	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
Full	84.12	0.99%	91.66	1.31%	95.51	1.54%	96.21	2.83%
Without MV	75.05	4.98%	81.96	11.82%	80.47	21.64%	81.52	7.84%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. "Full" refers to the full data set of variables. "No MV" refers to a slightly narrower data set where appraised market value and all associated variables such as market value per square foot, income return, capital appreciation return, cash flow return, and cap rate are removed. R^2s are expressed as a percentage.

interest amounts. However, for Model 3, while following the same flatline relationship as Model 1 for interest amounts below $400,000, it starts exhibiting a negative relationship for interest amounts above $400,000. This suggests that Model 3 believes that beyond a certain threshold, a large monthly interest payment may be a reflection of a distressed property.

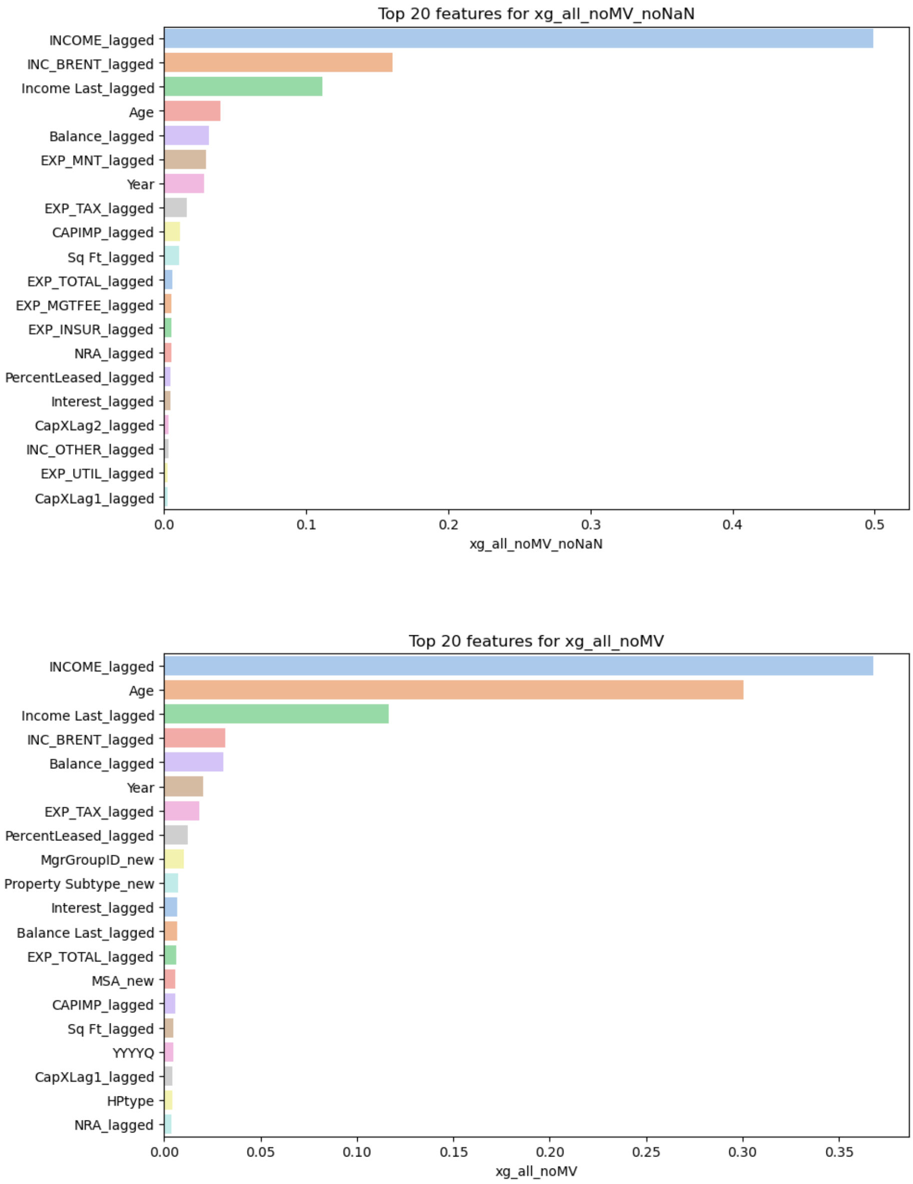
In summary, we demonstrate that researchers have to be careful in making conclusions when making use of data sets that contain complete information only. For top-ranked variables, marginal relationships can change from positive to negative once we start absorbing observations with incomplete information, suggesting that data may not be completely missing at random.

5.5 Modeling without market value related variables

We see in Figure 3 that appraised market values of properties play a dominant role in predicting prices, taking up more than 70 percent weight in terms of relative importance for both models. As a robustness test, it would be interesting to see how the models will react if we remove market values and related variables, such as market value per square foot, income return, capital appreciation return, cash flow return, and cap rate. It is not only an interesting study; such a model is important to investors or regulators as not all properties come with appraisal values. Such properties constitute 45.8 percent of the NCREIF database (see Table 1). Would using missing values improve predictive performance? What features or variables would then play an important role in predicting property values, when an important variable such as market valuation is missing?

Table 8 reports the findings. As expected, we see a big drop in R^2_{oos} after we exclude the market value and associated variables, from 84.12 percent to 75.05 percent in Model 1, which trains on observations with complete information. Can observations with missing values save the day? To some extent, yes. In Model 3, which trains on data starting from 2000, the R^2_{oos} increases from 75.05 percent to 80.47 percent. If we start training on incomplete data from 1978 in Model 4, we push the R^2_{oos} slightly further, up to 81.52 percent.

Figure 6: Top-20 most influential variables (without market value related variables)



Notes: Variable importance for the top-20 most influential variables in each model. The top panel is Model 1, which is trained on observations without missing values. The bottom panel is Model 3, which is trained on observations that include missing values. Variable importance within each model is normalized to sum to one.

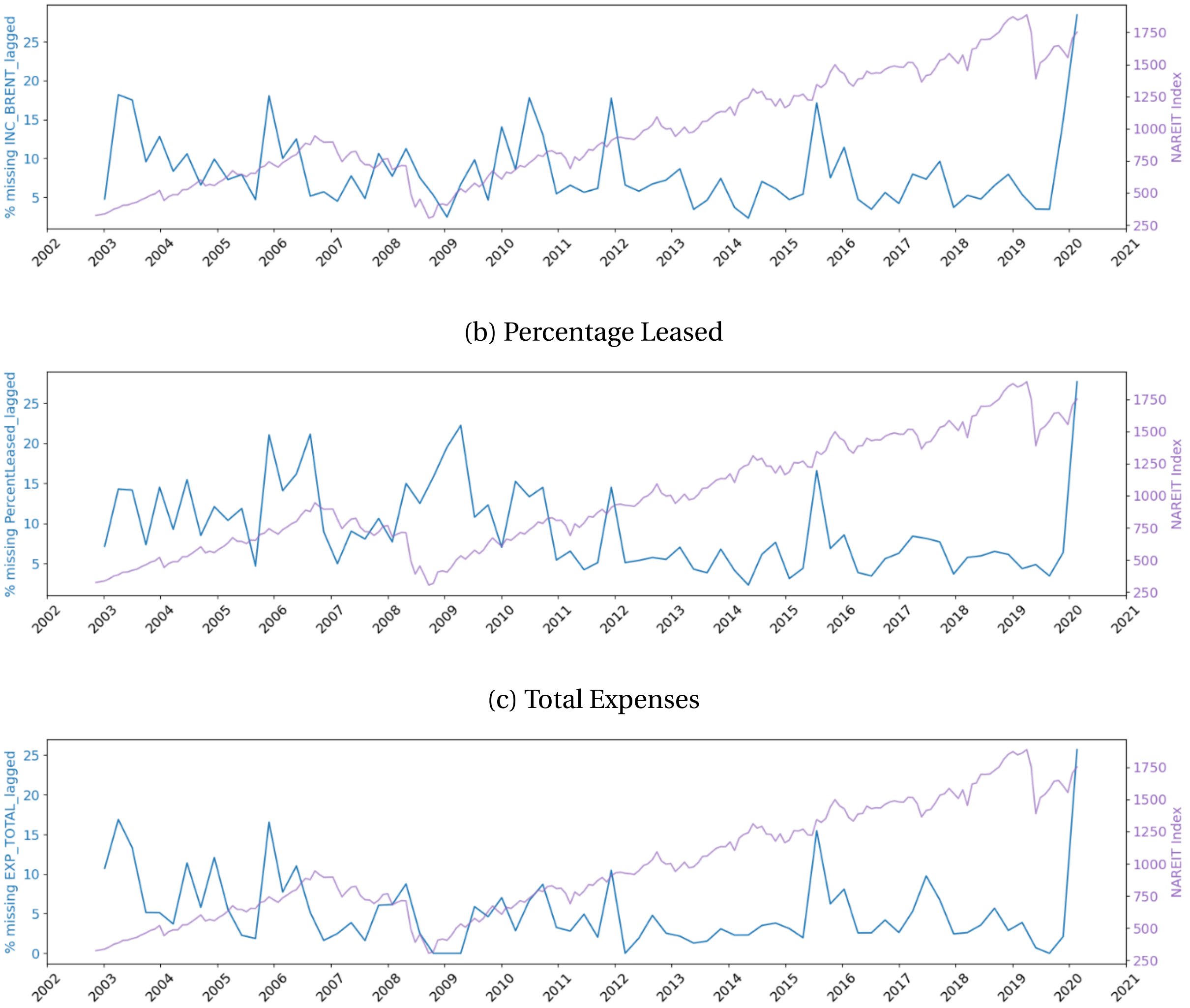
Looking at Figure 6, *NOI* now plays the largest role for both models, albeit at lower weight than *Market Value* used to take. For Model 1, *NOI* contributes approximately 50 percent to predictive ability, while for Model 3 it contributes approximately 35 percent. We also observe the same phenomenon for the full data set, where categorical variables such as property type, manager group ID, and MSA tend to rank as the most important variables for Model 3 but are missing in the Top-20 ranking for Model 1.

In summary, machine learning models are capable of adapting when important variables such as market valuation and cap rate are missing from observations, although one has to be satisfied with a lower R^2 of 80 percent. Second, even with fewer key variables in the picture, it is still important for models to include observations with missing values, as they can increase R^2s by 5-6 percentage points over Model 1's.

6 Are data missing at random?

Figure 7: Percentage of missing values over time

(a) Base Rent



(b) Percentage Leased

(c) Total Expenses

Notes: This figure plots the time series of missing values as a percentage of total observations in each calendar quarter from 2003 through 2020. The top panel plots percentage of missing values for base rent. The middle panel plots the percentage of missing values for leasing percentage. The bottom panel plots the percentage of missing values for total expenses. The three panels are overlaid with the time series of the NAREIT index (in purple). Numbers on the left-axis are expressed as a percentage, while numbers on the right-axis are expressed as index levels.

Although the process generating missing values in the NCREIF database is unobservable, we can still detect patterns in the occurrences of NaNs. Figure 7 plots the time series of missing values as a percentage of total observations in each calendar quarter for three predictors: base rent, percentage leased and total expenses (in blue). We also overlay the NAREIT index (in purple) as an indicator of overall market performance.

There are two observations. First, within a single year, the change in the percentage of missing values can vary greatly from quarter to quarter, doubling and halving in some cases. Second, from year to year, we observe spikes in missing values during certain periods. For instance, we see a spike in mid-2003, early-2006, mid-2010, early 2012, mid-2015 and early-2020. If the data were missing truly at random, smoother lines would be expected in Figure 7. Hughes and Nichols (2025) document how "no news is bad news" when monitoring real estate loan performance and we wonder if the same applies to NCREIF members. Do they voluntarily submit more data in good years while withholding data in less favourable conditions? This remains a question for further research.

Table 9 compares differences in selected characteristics for properties that have, and do not have missing values, in Figure 7. We observe statistically significant differences: Properties that are missing information on percentage leased have lower market values, are smaller in size, exhibit lower total returns and lower cap rates. Properties that have missing information on base rent and total expenses have lower market values, are smaller in square footage, have lower total returns but higher cap rates.

In Figure 5, we see that for commercial properties younger than 20 years old, the relationship between price and age is negative if we incorporate observations with missing values into our training set. However, the relationship between price and age is positive if we exclude observations with missing values from our training set. This could set the stage for a biased model that unintentionally inflates price estimates for older properties and penalizes them. Table 10 further shows that there are clear differences between young and old properties when it comes to missing values, so the effect of a missing value for age is not limited to the direct contribution of age to any estimates. For example, older properties tend to have more missing values for *Base Rent, Total Expenses, Cap Rate* which might influence any estimate indirectly.

In sum, dropping observations due to missingness will not only increase the uncertainty around a point estimate but is likely to introduce biases.

Table 10: NaN share (%) by Property Age

Variable	Age ≤ 20 years	Age 20+ years	Diff.
NOI	4.4	4.1	-0.3
NOI_Lag1	5.9	6.5	0.6
CapEx	9.4	7.4	-2.0
CapEx_Lag1	8.6	7.8	-0.8
CapEx_Lag2	10.9	9.9	-1.0
Market Value	4.4	4.1	-0.3
Market Value_Lag1	4.4	4.1	-0.3
Market Value_Lag2	5.9	6.5	0.6
Market Value per Sq Ft	18.5	10.2	-8.3
Market Value per Unit	63.5	78.1	14.6
Income Return	4.4	4.1	-0.3

Note: This table presents the percentage of NaNs for properties in the NCREIF database that are younger than 20 years old and older than 20 years old.

Table 10: NaN share (%) by Property Age (continued)

Variable	Age ≤ 20 years	Age 20+ years	Diff.
Capital Appreciation Return	4.4	4.1	-0.3
Total Return	4.4	4.1	-0.3
Cash Flow Return	4.4	4.1	-0.3
Lev. Income Return	4.4	4.1	-0.3
Lev. Appreciation Return	4.4	4.1	-0.3
Lev. Total Return	4.4	4.1	-0.3
Interest Payment	4.7	4.2	-0.5
Principal Payment	4.5	4.3	-0.2
Loan Balance	4.4	4.1	-0.3
Loan Balance_Lag1	4.4	4.1	-0.3
Loan Proceeds	4.5	4.2	-0.3
Sq Ft	18.0	9.8	-8.2
Units	59.1	73.3	14.2
Percentage Leased	7.4	9.1	1.7
Net Rentable Area	16.1	21.0	4.9
Additional Acquisition Costs	5.9	4.7	-1.2
Leasing Commissions	5.3	5.4	0.1
Tenant Improvements	6.5	6.4	-0.1
Building Improvements	9.0	7.7	-1.3
Bulding Expansion	5.0	4.5	-0.5
Other CapEx	6.9	6.0	-0.9
Other Principal Payment	4.5	4.2	-0.3
Regular Principal Payment	4.4	4.3	-0.1
Base Rent	7.8	25.8	18.0
Contingent Income	4.6	4.3	-0.3
Reimbursement Income	6.2	6.1	-0.1
Other Income	8.9	7.8	-1.1
Admin Expense	6.5	5.8	-0.7
Marketing Expense	5.4	5.1	-0.3
Utility Expense	5.6	4.9	-0.7
Maintenance Expense	5.1	4.7	-0.4
Insurance Expense	5.4	4.7	-0.7
Management Fee Expense	4.6	4.3	-0.3
Tax Expense	5.6	4.6	-1.0
Other Expense	7.3	8.3	1.0
Total Expense	4.9	23.9	19.0
Cap Rate	31.9	45.2	13.3

Note: This table presents the percentage of NaNs for properties in the NCREIF database that are younger than 20 years old and older than 20 years old.

Table 9: Characteristics of properties with and without missing values

	Panel A: Base Rent			
	Missing	Not Missing	t-statistic	p-value
Market Value ($000s)	15,626	40,434	-21.81	0.00
Sq Ft (000s)	220.46	260.79	-6.15	0.00
Total Return (%)	1.96	2.48	-3.37	0.00
Cap Rate (%)	6.90	6.14	6.65	0.00
	Panel B: Percentage Leased			
	Missing	Not Missing	t-statistic	p-value
Market Value ($000s)	12,604	36,051	-13.87	0.00
Sq Ft (000s)	166.94	258.60	-9.34	0.00
Total Return (%)	0.61	2.54	-8.47	0.00
Cap Rate (%)	4.20	6.47	-13.52	0.00
	Panel C: Total Expenses			
	Missing	Not Missing	t-statistic	p-value
Market Value ($000s)	15,621	39,754	-20.66	0.00
Sq Ft (000s)	220.93	259.55	-5.75	0.00
Total Return (%)	2.17	2.40	-1.44	0.15
Cap Rate (%)	8.69	5.78	25.02	0.00

Notes: This table reports the characteristics of properties with and without missing values in base rent (top panel), in leasing percentage (middle panel), and in total expenses (bottom panel). The measured characteristics are market values, square footage, total return, and cap rate.

7 Conclusion

This paper makes a case that missing data should be *used* and not *dropped*, especially in domains where data are scarce such as commercial real estate. Before the proliferation of machine learning approaches, missing values were a painful reality of empirical works. With algorithms such as sparsity awareness, machine learning models have been able to ease this challenge and deal with sparse data in fields ranging from image recognition and high-energy physics to real estate finance and economics. To demonstrate the often neglected opportunity of learning from incomplete observations that feature some missing data, we use a widely available algorithm, `XGBoost` (Chen and Guestrin, 2016), and apply it to the canonical NCREIF data. We find an improvement in out-of-sample predictability when we keep observations with missing values in our training samples when conducting a ‘fair horse race’ with more limited traditional models.

However, with machine learning, one does not need to ‘play fair’, as we can use data sets that are wider and longer, for example going back further into time (in the case of U.S. commercial real estate, a few decades earlier to 1978) to train models and learn from data, or accepting more covariates even if these fields may be sparsely populated. In danger of simplification, any information is good information when training ML models. This is in contrast to traditional linear models, which are compelled to train on complete observations, missing out on a huge opportunity to absorb bits and pieces of information from incomplete observations that are commonplace in the real estate world.

By looking at the marginal effects of asset-level variables on expected property values (or prices), we demonstrate that researchers must be cautious when drawing conclusions if they only make use of data sets that contain complete information. For top-ranked variables, marginal relationships can change from positive to negative (or vice versa) once we start absorbing observations with incomplete information, suggesting that the data may not be completely missing at random. For governments and regulators, it may lead to wrong policy-making or biases against certain segments of society.

In sum, dropping incomplete observations from a sample is wasteful. Data scientists and econometricians have developed the ML tools needed to squeeze more insights even from sparsely populated data, and real estate researchers should use them whenever data are scarce.

References

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Cannon, S. E. and Cole, R. (2011). How accurate are commercial real estate appraisals? evidence from 25 years of ncreif sales data. *Journal of Portfolio Management*, 37(5):68–88.

Chan, F., Schulz, R., and Zhang, Z. (2023). An application of machine learning in real estate economics: what extra benefits could machine learning techniques provide? In *Proceedings of the 25th International Congress on Modelling and Simulation, MODSIM 2023*. Modelling and Simulation Society of Australia and New Zealand Inc.(MSSANZ).

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794.

Deppner, J., von Ahlefeldt-Dehn, B., Beracha, E., and Schaefers, W. (2023). Boosting the accuracy of commercial real estate appraisals: An interpretable machine learning approach. *Journal of Real Estate Finance and Economics*. Advance online publication.

Elkind, D., Kaminski, K., Lo, A., Siah, K. W., and Wong, C. H. (2022). When do investors freak out? machine learning predictions of panic selling. *Journal of Financial Data Science*, 4(1):11–39.

Freyberger, J., Hoeppner, B., Neuhierl, A., and Weber, M. (2025). Missing data in asset pricing panels. *Review of Financial Studies*, 38(3):760–802.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies*, 33(5):2223–2273.

Heckman, J. J. (1976). The common structure of statistical models of truncation, sample selection, and limited dependent variables. *Annals of Economic and Social Measurement*, 5(4):475–492.

Heckman, J. J. (1979). Sample selection bias as a specification error. *Econometrica*, 47(1):153–161.

Ho, W. K. O., Tang, B., and Wong, S. W. (2021). Predicting property prices with machine learning algorithms. *Journal of Property Research*, 38(1):48–70.

Hughes, S. and Nichols, J. (2025). No news is bad news: Monitoring, risk, and stale financial performance in commercial real estate. *FEDS Working Paper*.

Leow, K. and Lindenthal, T. (2025). Enhancing real estate investment trust return forecasts using machine learning. *Real Estate Economics*.

LeSage, J. P. and Pace, R. K. (2004). Models for spatially dependent missing data. *Journal of Real Estate Finance and Economics*, 29(2):233–254.

Little, R. and Rubin, D. (2019). *Statistical Analysis with Missing Data*. Wiley, Hoboken, NJ, 3 edition.

Liu, T., Wei, H., and Zhang, K. (2018). Wind power prediction with missing data using gaussian process regression and multiple imputations. *Applied Soft Computing*, 71:905–916.

Park, J., Müller, J., Arora, B., Faybishenko, B., Pastorello, G., Varadharajan, C., Sahu, R., and Agarwal, D. (2023). Long-term missing value imputation for time series data using deep neural networks. *Neural Computing and Applications*, 35(12):9071–9091.

Peng, C. J., Harwell, M., Liou, S., and Ehman, L. H. (2006). Advances in missing data methods and implications for educational research. In Sawilowsky, S., editor, *Real Data Analysis*, pages 31–78.

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, 63(3):581–592.

Tabachnick, B. G. and Fidell, L. S. (2007). *Using Multivariate Statistics*. Pearson, 5 edition.

Thirukumaran, S. and Sumathi, A. (2016). Proceedings of the 10th international conference on intelligent systems and control (isco 2016). In *Proceedings of ISCO 2016*.

Warga, A. (1992). Bond returns, liquidity and missing data. *Journal of Financial and Quantitative Analysis*, 27(4):605–617.

Zaytar, M. A. and El Amrani, C. (2016). Sequence-to-sequence weather forecasting with long short-term memory recurrent neural networks. *International Journal of Computer Applications*, 143(11):7–11.

Zhou, L. and Lai, K. K. (2017). Adaboost models for corporate bankruptcy prediction with missing data. *Computational Economics*, 50(1):69–94.

A Description statistics of NCREIF data set

Table A.1: Descriptive Statistics of Numerical Variables

Variable	Mean	Median	Stdev	Min	Max
Year	2007.23	2008	9.12	1978.00	2020
Age	32.4	32	15.98	0.00	180
Sq Ft (000s)	249.46	173.94	334.5	0.06	20797.42
Units	271.97	263.5	178.63	1.00	3870
Percentage Leased (%)	90.29	95	14.23	0.09	100
Net Rentable Area (000s)	234.81	170.44	325.24	0.00	20797.42
Sale Price ($000s)	34,993	17,893	63,306	10.00	2,133,497
Market Value ($000s)	33,657	17,300	60,643	100.00	2,080,000
Market Value_Lag1 ($000s)	33,242	17,121	60,262	100.00	2,087,000
Market Value_Lag2 ($000s)	33,030	17,020	59,750	100.00	2,083,000
Market Value per Sq Ft	137.19	91.65	158.99	0.71	2718.26
Market Value per Unit ($000s)	155.07	114.88	146.53	8.08	3,045.75
Cap Rate (%)	6.29	6.22	4.08	-28.11	29.91
NOI ($000s)	502.23	287.7	866.24	-8528.79	24919.11
NOI_Lag1 ($000s)	496.88	283.76	867.22	-3110.16	26772.14
Base Rent ($000s)	846.44	536.23	1350.17	0.11	40246.18
Contingent Income ($000s)	2.41	0	40.62	0.00	3326.93
Reimbursement Income ($000s)	91.87	5.77	337.61	0.00	15869.33
Other Income ($000s)	53.38	0.19	312.54	0.00	10372.94
CapEx ($000s)	190.53	25.25	1040.63	0.00	86723.96
CapEx_Lag1 ($000s)	252.07	25.04	1924.91	0.00	100706.87
CapEx_Lag2 ($000s)	329.87	26.91	2776	0.00	174763.67
Additional Acq Costs ($000s)	143.23	0	2820.04	0.00	221740.22
Leasing Commissions ($000s)	28.16	0	150.54	0.00	7946.93
Tentant Improvements ($000s)	69.79	0	702.8	0.00	69819.59
Building Improvements ($000s)	87.58	0	1168.81	0.00	71477.32
Building Expansion ($000s)	2.68	0	81.24	0.00	6474.58
Other CapEx ($000s)	20.82	0	713.73	0.00	80963.13
Income Return (%)	1.71	1.69	1.44	-26.13	53.77
Capital Appreciation Return (%)	0.63	0	8.01	-80.35	112.23
Total Return (%)	2.34	1.9	8.14	-80.27	112.88
Cash Flow Return (%)	1.12	1.41	2.53	-80.65	53.62
Lev. Income Return (%)	1.83	1.85	20.03	-1079.91	878.18
Lev. Appreciation Return (%)	-0.87	0	188.35	-21328.90	1871.42
Lev. Total Return (%)	0.96	2.07	184.92	-20873.74	2291.53
Interest Payment ($000s)	120.82	0	358.06	0.00	14786.82
Principal Payment ($000s)	189.73	0	2289.04	0.00	130528.19
Regular Principal Payment ($000s)	75.02	0	1480.18	0.00	130528.19
Other Principal Payment ($000s)	114.53	0	1744.44	0.00	89705.13

Notes: This table presents the summary statistics of numerical variables found in the NCREIF database. All variables are lagged by 2 calendar quarters for robustness purposes. For example, "Market Value", "Market Value_Lag1" and "Market Value_Lag2" refer to market value lagged by 6 months, 9 months and 12 months, respectively.

Table A.1: Descriptive Statistics of Numerical Variables (continued)

Variable	Mean	Median	Stdev	Min	Max
Loan Balance ($000s)	9794	0	26760	0.00	950000
Loan Balance_Lag1 ($000s)	9837	0	26685	0.00	950000
New Financing ($000s)	131.9	0	2687.62	0.00	161500
Admin Expense ($000s)	38.54	3.02	105.56	0.00	4434.48
Marketing Expense ($000s)	11.84	0	55.78	0.00	2092.79
Utility Expense ($000s)	40.62	7.61	117.08	0.00	8762.33
Maintenance Expense ($000s)	65.35	17.3	211.7	0.00	16772.73
Insurance Expense ($000s)	12.72	4.63	31.88	0.00	1660.57
Management Fee Expense ($000s)	21.52	8.94	88.99	0.00	9241.54
Tax Expense ($000s)	97.06	40.09	237.04	0.00	10929.75
Other Expense ($000s)	54.94	2.01	362.63	0.00	17099.37
Total Expense ($000s)	449.76	245.75	901.24	0.07	26577.01

Notes: This table presents the summary statistics of numerical variables found in the NCREIF database. All variables are lagged by 2 calendar quarters for robustness purposes. For example, "Market Value", "Market Value_Lag1" and "Market Value_Lag2" refer to market value lagged by 6 months, 9 months and 12 months, respectively.

Table A.2: Descriptive Statistics of Dummy and Categorical Variables

Variable	Count	Percentage
Joint Venture		
No	10981	75.9
Yes	3489	24.1
Has Leverage		
No	8558	59.1
Yes	5912	40.9
Post NPI Freeze		
No	4081	28.2
Yes	10389	71.8
YYYYQ		
20052	289	2.0
20193	277	1.9
20144	260	1.8
.	.	.
.	.	.
.	.	.
19802	1	0

Notes: This table presents the summary statistics of dummy and categorical variables found in the NCREIF database.

Table A.2: Descriptive Statistics of Dummy and Categorical Variables (continued)

Variable	Count	Percentage
19812	1	0
19781	1	0
Property Type		
Apartment	3402	23.5
Hotel	289	2.0
Industrial	4964	34.3
Office	3738	25.8
Retail	2077	14.4
Property Subtype		
Apartment - Garden	2158	14.9
Apartment - High Rise	774	5.3
Apartment - Low Rise	236	1.6
Industrial - R&D	480	3.3
Industrial - Flex Space	652	4.5
Industrial - Manufacturing	30	0.2
Industrial - Other	114	0.8
Industrial - Office Showroom	20	0.1
Industrial - Warehouse	3668	25.3
Office - CBD	720	5.0
Office - Suburban	3018	20.9
Retail - Community	654	4.5
Retail - Theme / Festival Center	4	0.0
Retail - Fashion / Specialty Center	47	0.3
Retail - Neighborhood	756	5.2
Retail - Outlet	3	0.0
Retail - Power Center	141	1.0
Retail - Regional	156	1.1
Retail - Super Regional	89	0.6
Retail - Single Tenant	215	1.5
NaN	535	3.7
Manager Group ID		
75	1176	8.1
71	1117	7.7
19	746	5.2
.	.	.
.	.	.
.	.	.
73	27	0.2

Notes: This table presents the summary statistics of dummy and categorical variables found in the NCREIF database.

Table A.2: Descriptive Statistics of Dummy and Categorical Variables (continued)

Variable	Count	Percentage
14	22	0.2
157	13	0.1
Others	1341	9.3
MSA		
4472	1323	9.1
1602	955	6.6
1922	953	6.6
.	.	.
.	.	.
.	.	.
5880	28	0.2
4520	24	0.2
3120	22	0.2
Others	954	6.6
Region		
East	3177	22
Midwest	2276	15.7
South	4355	30.1
West	4662	32.2
Division		
East North Central	1635	11.3
Mideast	1550	10.7
Northeast	1627	11.2
Southeast	2371	16.4
Southwest	1984	13.7
West Mountain	1288	8.9
West North Central	641	4.4
West Pacific	3374	23.3
Appraisal		
None	6610	45.7
Internal	5030	34.8
External	2815	19.5
NaN	15	0.1
FundType		
Separate Account	3271	22.6
ODCE Fund	1882	13.0
Closed End	1352	9.3

Notes: This table presents the summary statistics of dummy and categorical variables found in the NCREIF database.

Table A.2: Descriptive Statistics of Dummy and Categorical Variables (continued)

Variable	Count	Percentage
Open End	1026	7.1
Not Elsewhere Classified	77	0.5
Public REIT	1	0.0
NaN	6861	47.4

Notes: This table presents the summary statistics of dummy and categorical variables found in the NCREIF database.

B Machine learning algorithm

Figure B.1: Sparsity-aware Split Finding Algorithm (Chen and Guestrin 2016)

```
**Input**: $I$, instance set of current node
**Input**: $I_k = \{i \in I | x_{ik} \neq \text{missing}\}$
**Input**: $d$, feature dimension
*Also applies to the approximate setting, only collect*
*statistics of non-missing entries into buckets*
$gain \leftarrow 0$
$G \leftarrow \sum_{i \in I}, g_i, H \leftarrow \sum_{i \in I} h_i$
for $k = 1$ to $m$ do
    // enumerate missing value goto right
    $G_L \leftarrow 0,\ H_L \leftarrow 0$
    for $j$ in sorted($I_k$, ascent order by $\mathbf{x}_{jk}$) do
        $G_L \leftarrow G_L + g_j,\ H_L \leftarrow H_L + h_j$
        $G_R \leftarrow G - G_L,\ H_R \leftarrow H - H_L$
        $score \leftarrow \max(score, \frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{G^2}{H+\lambda})$
    end
    // enumerate missing value goto left
    $G_R \leftarrow 0,\ H_R \leftarrow 0$
    for $j$ in sorted($I_k$, descent order by $\mathbf{x}_{jk}$) do
        $G_R \leftarrow G_R + g_j,\ H_R \leftarrow H_R + h_j$
        $G_L \leftarrow G - G_R,\ H_L \leftarrow H - H_R$
        $score \leftarrow \max(score, \frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{G^2}{H+\lambda})$
    end
end
**Output**: Split and default directions with max gain
```

C Default hyperparameters for machine learning method

We do not require a validation sample as we do not perform any hyperparameter optimization, following Elkind et al. (2022). We employ `XGBoost`[6], which stands for Extreme Gradient Boosting. It is a scalable, distributed gradient-boosted regression tree (GBRT) machine learning library developed by Chen and Guestrin (2016). It provides parallel tree boosting and is the leading machine learning library for regression, classification, and ranking problems.

We use default hyperparameters for `XGBoost`. This forms the lowest bound of performance for our machine learning models. Machine learning training is executed on an Apple M1 Ultra chip with a 20-core CPU, a 48-core GPU and 128 GB unified memory.[7]

Table C.1: Default hyperparameters for machine learning method

No.	Machine Learning Model	Default Hyperparameters

1	`XGBoost`	n_trees=100
		learning_rate=0.3
		min_split_loss=0
		max_depth=6
		min_child_weight=1
		subsample=1
		sampling_method=uniform
		l1_regulartization=0
		l2_regulartization=1
		tree_method=auto

[6] xgboost v1.3.3 , https://xgboost.readthedocs.io/en/stable/

[7] While these CPU, GPU and memory specifications are extremely powerful for a personal computer (Apple claims the M1 Ultra is the most powerful chip ever in a personal computer, as of 1 April 2023), regression trees do stretch the computer to its limit, even without attempting hyperparameter tuning. Equipped with a more powerful GPU such as the Nvidia Tesla K80 with thousands of cores, hyperparameter tuning can take place and we would expect better performance results for NCREIF property values, but we do not expect a qualitative difference in our conclusions.

D Yearly breakdown of transactions, by property type

Table D.1: Yearly breakdown of transactions with and without missing values (Industrial)

Year	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*	*Year*	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*
1978	2	0	-100	2000	84	6	-92.9
1979	1	0	-100	2001	83	13	-84.3
1980	3	0	-100	2002	113	28	-75.2
1981	3	0	-100	2003	123	21	-82.9
1982	12	0	-100	2004	223	25	-88.8
1983	25	0	-100	2005	213	31	-85.4
1984	33	0	-100	2006	230	40	-82.6
1985	61	0	-100	2007	206	38	-81.6
1986	61	0	-100	2008	76	13	-82.9
1987	45	0	-100	2009	91	20	-78.0
1988	68	0	-100	2010	107	33	-69.2
1989	74	0	-100	2011	126	24	-81.0
1990	51	0	-100	2012	248	66	-73.4
1991	53	0	-100	2013	301	82	-72.8
1992	39	0	-100	2014	236	58	-75.4
1993	42	0	-100	2015	196	73	-62.8
1994	79	0	-100	2016	247	104	-57.9
1995	58	0	-100	2017	241	56	-76.8
1996	121	0	-100	2018	145	34	-76.6
1997	157	0	-100	2019	272	111	-59.2
1998	111	0	-100	2020	212	55	-74.1
1999	92	0	-100	Total	4964	931	-81.2

Notes: This table reports the yearly breakdown of NCREIF sale transactions, with and without missing values. Columns 3 and 7 display the number of observations that do not contain missing values within the 63 data fields shown in Tables 1 and 2, with the exclusion of Units and Market Value per Unit.

Table D.2: Yearly breakdown of transactions with and without missing values (Office)

Year	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*	*Year*	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*
1978	1	0	-100	2000	109	9	-91.7
1979	0	0	N.A.	2001	64	4	-93.8
1980	0	0	N.A.	2002	86	16	-81.4
1981	1	0	-100	2003	120	24	-80.0
1982	3	0	-100	2004	167	37	-77.8
1983	10	0	-100	2005	219	36	-83.6
1984	27	0	-100	2006	194	46	-76.3
1985	18	0	-100	2007	199	43	-78.4
1986	33	0	-100	2008	79	18	-77.2
1987	26	0	-100	2009	57	15	-73.7
1988	27	0	-100	2010	69	13	-81.2
1989	50	0	-100	2011	70	21	-70.0
1990	42	0	-100	2012	130	48	-63.1
1991	38	0	-100	2013	167	65	-61.1
1992	31	0	-100	2014	170	57	-66.5
1993	49	0	-100	2015	161	64	-60.2
1994	40	0	-100	2016	190	75	-60.5
1995	58	0	-100	2017	164	46	-72.0
1996	111	0	-100	2018	206	73	-64.6
1997	101	0	-100	2019	160	62	-61.3
1998	118	0	-100	2020	88	31	-64.8
1999	85	0	-100	Total	3738	803	-78.5

Notes: This table reports the yearly breakdown of NCREIF sale transactions, with and without missing values. Columns 3 and 7 display the number of observations that do not contain missing values within the 63 data fields shown in Tables 1 and 2, with the exclusion of Units and Market Value per Unit.

Table D.3: Yearly breakdown of transactions with and without missing values (Apartment)

Year	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*	*Year*	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*
1978	1	0	-100	2000	69	7	-89.9
1979	1	0	-100	2001	97	12	-87.6
1980	0	0	N.A.	2002	73	17	-76.7
1981	0	0	N.A.	2003	89	25	-71.9
1982	0	0	N.A.	2004	115	27	-76.5
1983	1	0	-100	2005	162	39	-75.9
1984	0	0	N.A.	2006	180	30	-83.3
1985	2	0	-100	2007	128	40	-68.8
1986	2	0	-100	2008	73	14	-80.8
1987	1	0	-100	2009	84	19	-77.4
1988	9	0	-100	2010	90	20	-77.8
1989	5	0	-100	2011	135	39	-71.1
1990	5	0	-100	2012	173	68	-60.7
1991	5	0	-100	2013	232	114	-50.9
1992	10	0	-100	2014	179	80	-55.3
1993	39	0	-100	2015	182	80	-56.0
1994	37	0	-100	2016	251	125	-50.2
1995	32	1	-96.9	2017	197	51	-74.1
1996	59	0	-100	2018	183	45	-75.4
1997	79	0	-100	2019	178	52	-70.8
1998	70	0	-100	2020	117	47	-59.8
1999	57	0	-100	Total	3402	952	-72.0

Notes: This table reports the yearly breakdown of NCREIF sale transactions, with and without missing values. Columns 3 and 7 display the number of observations that do not contain missing values within the 63 data fields shown in Tables 1 and 2, with the exclusion of Units and Market Value per Unit.

Table D.4: Yearly breakdown of transactions with and without missing values (Retail)

Year	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*	*Year*	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*
1978	1	0	-100	2000	51	5	-90.2
1979	0	0	N.A.	2001	55	5	-90.9
1980	0	0	N.A.	2002	56	10	-82.1
1981	0	0	N.A.	2003	75	9	-88.0
1982	7	0	-100	2004	95	12	-87.4
1983	11	0	-100	2005	162	41	-74.7
1984	15	0	-100	2006	51	11	-78.4
1985	16	0	-100	2007	62	15	-75.8
1986	27	0	-100	2008	17	4	-76.5
1987	18	0	-100	2009	19	2	-89.5
1988	24	0	-100	2010	47	6	-87.2
1989	19	0	-100	2011	66	22	-66.7
1990	10	0	-100	2012	59	20	-66.1
1991	9	0	-100	2013	142	49	-65.5
1992	17	0	-100	2014	95	25	-73.7
1993	24	0	-100	2015	93	31	-66.7
1994	24	0	-100	2016	87	40	-54.0
1995	35	0	-100	2017	67	19	-71.6
1996	65	0	-100	2018	36	8	-77.8
1997	112	0	-100	2019	70	31	-55.7
1998	98	0	-100	2020	55	12	-78.2
1999	85	0	-100	Total	2077	377	-81.8

Notes: This table reports the yearly breakdown of NCREIF sale transactions, with and without missing values. Columns 3 and 7 display the number of observations that do not contain missing values within the 63 data fields shown in Tables 1 and 2, with the exclusion of Units and Market Value per Unit.

Table D.5: Yearly breakdown of transactions with and without missing values (Hotel)

Year	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*	*Year*	*Total No. of Obs.*	*Obs. with no NaNs*	*Data Loss (%)*
1978	0	0	N.A.	2000	5	0	-100.0
1979	0	0	N.A.	2001	3	0	-100.0
1980	0	0	N.A.	2002	6	0	-100.0
1981	0	0	N.A.	2003	10	0	-100.0
1982	1	0	-100	2004	7	0	-100.0
1983	0	0	N.A.	2005	15	1	-93.3
1984	1	0	-100	2006	9	0	-100.0
1985	0	0	N.A.	2007	8	1	-87.5
1986	1	0	-100	2008	6	2	-66.7
1987	1	0	-100	2009	1	0	-100.0
1988	1	0	-100	2010	4	0	-100.0
1989	1	0	-100	2011	13	3	-76.9
1990	2	0	-100	2012	9	4	-55.6
1991	2	0	-100	2013	21	13	-38.1
1992	0	0	N.A.	2014	90	71	-21.1
1993	2	0	-100	2015	9	5	-44.4
1994	2	0	-100	2016	9	6	-33.3
1995	2	0	-100	2017	7	3	-57.1
1996	14	0	-100	2018	11	2	-81.8
1997	3	0	-100	2019	10	7	-30.0
1998	2	0	-100	2020	1	0	-100.0
1999	0	0	N.A.	Total	289	118	-59.2

Notes: This table reports the yearly breakdown of NCREIF sale transactions, with and without missing values. Columns 3 and 7 display the number of observations that do not contain missing values within the 63 data fields shown in Tables 1 and 2, with the exclusion of Units and Market Value per Unit.

E Out of sample R^2 and MPE by year (natural log of variables)

Table E.1: Out of sample R^2 and MPE by year, on natural log of variables

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Train Set	No NaNs		With NaNs		With NaNs		With NaNs	
Test Set	No NaNs		With NaNs		No NaNs		No NaNs	
Train Start	2000		2000		2000		1978	
Test Start	2001		2001		2001		2001	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	51.02	-0.05%	88.08	-0.20%	94.64	0.16%	97.40	0.03%
2002	93.04	-0.10%	86.48	0.04%	94.49	0.15%	95.47	0.11%
2003	89.65	0.04%	88.27	0.19%	88.58	-0.08%	88.98	-0.05%
2004	92.41	0.39%	93.7	0.42%	95.84	0.22%	96.82	0.07%
2005	93.83	0.55%	90.26	0.39%	94.32	0.42%	95.23	0.41%
2006	97.28	0.13%	89.01	0.35%	96.61	0.07%	97.19	0.09%
2007	95.85	-0.07%	94.37	-0.02%	95.98	-0.23%	96.88	-0.10%
2008	98.24	-0.48%	91.42	-0.99%	97.58	-0.52%	98.27	-0.49%
2009	92.00	-1.10%	89.54	-0.47%	93.09	-0.31%	94.44	-0.68%
2010	96.45	0.24%	91.10	0.76%	94.74	0.67%	95.43	0.60%
2011	90.10	0.58%	83.71	0.33%	94.06	0.48%	92.94	0.54%
2012	96.66	-0.26%	95.08	-0.14%	96.69	-0.13%	96.17	-0.08%
2013	97.41	0.14%	94.97	0.15%	97.82	0.13%	97.30	0.09%
2014	95.46	0.14%	95.05	0.33%	90.55	0.55%	94.56	0.37%
2015	97.20	0.13%	96.41	0.10%	97.75	0.05%	97.49	0.12%
2016	98.30	-0.12%	90.73	-0.30%	98.21	-0.06%	98.44	-0.10%
2017	98.16	-0.17%	96.24	0.01%	98.33	-0.17%	98.37	-0.15%
2018	97.79	-0.01%	96.27	0.02%	98.28	-0.03%	98.03	-0.03%
2019	98.78	0.05%	97.28	0.18%	98.73	0.04%	98.83	0.00%
2020	98.42	0.19%	86.52	-0.70%	98.47	0.17%	98.58	0.13%
All Years	96.76	0.05%	92.90	0.07%	96.95	0.09%	97.29	0.07%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

F Out of sample R^2 and MPE by year, split by property type

Table F.1: Out of sample R^2 and MPE by year (Industrial)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	-24.52	40.88%	74.93	-0.28%	80.57	-0.01%	91.10	-0.18%
2002	84.33	-5.35%	29.20	23.08%	84.33	2.00%	82.57	-9.31%
2003	67.93	2.68%	64.45	7.09%	82.46	-1.73%	87.50	1.38%
2004	31.24	27.07%	74.44	8.24%	85.35	1.49%	86.54	1.67%
2005	89.15	16.53%	80.84	7.98%	85.92	4.07%	85.21	10.27%
2006	96.47	4.01%	59.82	7.86%	95.44	3.64%	94.91	2.49%
2007	97.28	1.66%	92.97	0.53%	93.39	-4.80%	91.57	2.60%
2008	93.65	-8.67%	81.28	-13.16%	69.12	-2.23%	85.77	-6.51%
2009	80.89	-17.33%	47.32	-14.10%	50.56	-9.15%	78.73	-10.73%
2010	34.32	15.65%	35.93	26.63%	40.50	15.94%	40.10	12.79%
2011	83.68	22.28%	-74.48	8.33%	-89.01	21.28%	-1189.57	16.97%
2012	93.54	-6.23%	89.92	-2.25%	95.42	-1.16%	95.32	-0.03%
2013	92.53	-0.82%	89.61	8.69%	94.70	0.92%	97.07	1.07%
2014	94.88	2.65%	92.57	3.56%	97.22	4.34%	97.84	5.24%
2015	92.49	3.60%	80.02	5.06%	95.27	3.44%	95.29	2.49%
2016	92.55	-0.86%	86.69	-5.75%	97.13	-2.84%	95.93	-1.83%
2017	95.31	-4.41%	78.61	7.07%	97.85	-3.54%	96.53	-5.59%
2018	97.17	3.55%	71.57	14.39%	96.6	6.10%	96.57	2.93%
2019	98.04	4.50%	95.63	2.20%	95.94	2.13%	95.19	4.75%
2020	96.52	7.05%	83.35	-17.05%	97.49	4.98%	97.09	7.91%
All Years	66.17	3.30%	69.16	3.73%	68.99	1.83%	62.63	2.15%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table F.2: Out of sample R^2 and MPE by year (Office)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	-375610.37	17.80%	85.89	21.19%	79.71	22.07%	69.38	-8.67%
2002	73.57	14.95%	-11.47	-5.46%	88.94	5.80%	92.70	40.54%
2003	82.18	-3.37%	81.29	3.99%	95.72	3.15%	84.84	5.03%
2004	58.80	15.12%	58.65	14.30%	98.16	2.11%	98.11	-1.77%
2005	83.24	17.76%	94.01	14.79%	90.79	7.58%	89.35	10.16%
2006	92.05	6.19%	65.51	14.92%	95.69	-0.86%	94.93	1.38%
2007	93.98	3.79%	74.58	2.79%	94.6	-0.74%	95.11	-0.49%
2008	98.70	-3.77%	77.06	-12.08%	77.25	-4.11%	84.63	-7.74%
2009	51.75	-26.03%	66.23	1.83%	83.99	-3.98%	82.08	-119.54%
2010	97.63	-1.15%	92.73	12.36%	84.28	-3.54%	78.48	0.21%
2011	67.25	3.24%	76.96	8.40%	78.28	9.88%	86.31	15.64%
2012	77.18	-6.92%	95.97	-7.61%	94.89	0.64%	95.89	-1.08%
2013	89.52	-0.03%	96.31	3.35%	96.78	3.01%	95.04	3.68%
2014	94.77	2.08%	94.22	1.63%	97.61	0.73%	98.04	0.89%
2015	77.12	-0.96%	94.46	0.03%	92.75	-3.85%	94.37	-5.04%
2016	81.73	-4.52%	88.52	-5.16%	86.30	-2.84%	87.09	-6.41%
2017	97.91	-3.29%	90.22	5.06%	97.87	0.60%	97.14	-1.63%
2018	98.82	-1.87%	95.58	-7.38%	94.64	-26.70%	94.96	2.59%
2019	98.16	6.93%	97.12	6.95%	98.92	3.34%	97.52	6.79%
2020	91.30	4.26%	97.8	-3.97%	97.41	4.93%	97.63	3.24%
All Years	83.55	1.42%	88.65	3.61%	93.08	-1.50%	93.24	-0.57%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table F.3: Out of sample R^2 and MPE by year (Apartment)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	4.78	31.18%	76.48	-1.98%	92.80	6.67%	96.45	5.04%
2002	51.97	3.76%	89.69	2.73%	83.17	0.33%	85.04	-4.77%
2003	72.71	11.47%	74.30	2.68%	98.30	-1.08%	98.76	0.22%
2004	70.97	13.63%	79.47	-1.15%	92.10	-0.20%	92.59	2.41%
2005	23.02	24.78%	45.02	11.48%	33.55	13.31%	40.04	13.69%
2006	92.27	4.01%	73.58	10.86%	94.14	-0.94%	93.86	-0.36%
2007	94.32	-3.48%	92.46	-1.29%	90.90	-3.81%	91.69	-6.51%
2008	93.43	-5.75%	82.37	-11.66%	94.19	-8.29%	95.69	-7.43%
2009	45.92	-20.99%	82.38	-12.44%	79.82	-19.21%	58.80	-21.60%
2010	86.54	0.82%	93.43	7.00%	94.30	-5.39%	96.38	-0.99%
2011	96.23	1.81%	92.84	0.27%	97.90	0.96%	98.05	-2.51%
2012	95.91	-4.22%	88.08	-2.88%	93.66	-4.17%	93.62	-4.94%
2013	96.74	2.33%	91.19	3.69%	97.25	4.90%	97.39	1.10%
2014	95.72	0.60%	90.06	2.71%	90.96	3.46%	88.36	2.82%
2015	56.61	5.32%	77.74	10.51%	75.83	6.76%	74.90	5.68%
2016	86.92	-0.73%	93.86	1.20%	97.60	0.95%	97.79	0.49%
2017	94.89	0.76%	97.87	-1.52%	98.19	-1.02%	98.68	-0.47%
2018	97.28	1.57%	92.92	0.26%	98.07	0.52%	98.54	2.35%
2019	97.17	-2.16%	94.70	0.97%	95.79	-1.44%	96.30	-1.40%
2020	98.65	-1.75%	97.39	0.38%	98.51	0.84%	98.91	0.72%
All Years	75.92	2.04%	87.40	1.90%	81.76	1.05%	83.15	0.27%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table F.4: Out of sample R^2 and MPE by year (Retail)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	-28.92	229.37%	75.19	-0.04%	70.84	33.33%	86.59	9.03%
2002	87.68	16.54%	93.32	-2.91%	93.88	2.78%	93.74	4.96%
2003	67.07	152.84%	85.58	8.48%	97.45	-0.99%	92.31	1.72%
2004	-31.57	43.05%	87.14	9.40%	90.50	12.35%	94.90	9.66%
2005	90.79	3.05%	87.17	13.54%	94.21	13.54%	94.11	11.49%
2006	81.21	11.63%	96.56	2.79%	76.07	1.64%	88.46	5.13%
2007	89.42	3.90%	88.74	2.38%	93.41	-7.68%	95.78	-4.92%
2008	98.05	1.52%	81.49	-3.47%	73.43	-8.83%	72.51	-11.43%
2009	79.28	-11.17%	62.79	-20.16%	58.50	-29.89%	51.67	-12.72%
2010	98.36	11.06%	91.06	3.06%	90.20	10.12%	92.14	25.25%
2011	92.14	12.17%	93.43	3.39%	92.58	5.51%	93.84	8.68%
2012	90.47	-1.49%	74.52	-0.61%	95.97	-0.68%	97.57	-4.08%
2013	92.10	0.37%	88.00	15.67%	96.27	9.50%	97.59	-8.80%
2014	94.03	5.44%	87.28	8.00%	88.29	1.88%	95.22	2.90%
2015	94.39	0.20%	95.66	6.31%	96.95	7.54%	94.54	4.85%
2016	94.53	1.49%	90.97	4.67%	97.58	3.58%	95.78	3.18%
2017	60.77	-4.39%	95.43	-6.68%	64.77	-4.31%	66.65	-2.56%
2018	91.74	-9.20%	95.99	-6.55%	92.05	6.05%	91.68	-13.43%
2019	68.29	2.88%	96.13	4.66%	97.29	2.98%	97.79	-1.66%
2020	98.22	3.40%	94.92	-6.82%	95.93	6.74%	96.07	-0.21%
All Years	76.58	10.74%	89.36	4.96%	94.98	5.09%	95.84	1.55%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table F.5: Out of sample R^2 and MPE by year (Hotel)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2011	-62.62	-22.43%	88.40	81.68%	-225.81	284.93%	-185.32	187.75%
2012	38.80	5.74%	47.94	-1.99%	76.24	44.59%	73.54	61.95%
2013	81.04	-22.72%	91.13	12.57%	91.75	15.21%	95.08	17.87%
2014	85.87	0.61%	85.02	19.00%	74.15	22.40%	79.47	29.11%
2015	51.94	19.63%	-18.36	7.11%	91.27	11.34%	90.02	7.54%
2016	96.51	4.13%	94.00	-1.59%	97.25	0.28%	93.65	-11.02%
2017	72.35	13.19%	89.00	4.18%	85.62	15.65%	87.52	6.34%
2018	-713.19	56.33%	78.62	-1.53%	-153.30	22.79%	-171.70	23.29%
2019	75.69	-6.50%	74.27	-7.50%	82.63	-10.58%	-67.98	-12.32%
All Years	81.50	-0.59%	81.65	16.79%	88.88	25.42%	78.29	26.85%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

G Performance metrics for data sets with reduced number of covariates, by year and by property type

Table G.1: Out of sample R^2 and MPE by year (Reduced data set, All property types)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	95.14	-2.10%	91.09	-1.02%	95.22	-3.60%	93.48	-1.83%
2002	91.10	-0.56%	89.30	9.71%	91.59	1.82%	93.60	9.15%
2003	84.65	3.95%	88.46	-104.96%	88.78	-159.84%	86.80	4.91%
2004	94.27	5.38%	70.00	19.47%	97.35	4.63%	97.41	3.18%
2005	88.75	9.44%	86.84	9.54%	88.63	8.28%	89.70	8.26%
2006	88.77	2.74%	85.43	12.57%	88.88	6.35%	92.14	-0.19%
2007	73.92	-2.29%	79.64	-1.30%	77.32	-0.80%	74.98	0.04%
2008	84.76	-10.02%	85.02	-7.28%	83.94	-1.61%	82.58	-10.64%
2009	62.08	-12.05%	74.11	-7.13%	71.89	-6.44%	89.07	14.96%
2010	96.23	14.58%	94.58	14.17%	96.73	14.59%	91.09	35.25%
2011	96.67	12.99%	91.67	13.72%	96.23	7.28%	95.68	10.21%
2012	96.16	-0.55%	93.03	-1.59%	95.91	-1.10%	96.65	0.35%
2013	73.95	2.21%	95.85	0.03%	96.85	3.44%	95.66	-8.06%
2014	97.12	2.26%	89.16	3.05%	96.66	2.50%	96.86	4.76%
2015	92.77	2.49%	94.37	2.98%	94.38	0.32%	96.08	0.74%
2016	95.23	-2.35%	93.46	-3.49%	93.67	-5.12%	92.35	-4.65%
2017	97.28	-2.86%	90.25	2.70%	97.77	-1.05%	96.68	-2.72%
2018	97.92	0.32%	98.77	2.19%	99.37	2.43%	95.31	2.20%
2019	98.46	-1.65%	97.44	0.48%	97.95	-0.80%	97.93	-1.11%
2020	97.23	2.53%	96.00	-8.27%	97.36	1.27%	96.78	1.70%
All Years	92.49	1.55%	91.85	-0.68%	94.18	-4.26%	93.21	2.00%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table G.2: Out of sample R^2 and MPE by year (Reduced data set, Industrial only)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	80.75	1.81%	75.18	-4.07%	82.66	-0.63%	89.04	4.87%
2002	62.98	-5.43%	20.61	36.63%	67.91	-6.62%	73.11	-0.40%
2003	63.91	7.98%	64.68	5.75%	82.02	4.45%	88.54	5.17%
2004	87.00	8.29%	74.07	17.76%	74.85	9.15%	87.50	0.13%
2005	86.37	11.70%	80.37	7.38%	86.11	6.03%	89.39	10.66%
2006	56.89	1.81%	48.54	7.43%	9.36	4.73%	-28.34	4.37%
2007	81.17	-0.37%	57.15	11.20%	70.65	2.77%	74.28	1.58%
2008	92.65	-14.16%	85.48	-14.24%	85.73	-11.76%	91.30	-12.19%
2009	55.58	-18.56%	47.39	-15.52%	52.53	-16.74%	63.97	-14.91%
2010	47.40	16.61%	50.90	20.90%	51.61	16.80%	49.02	18.29%
2011	89.62	-32.30%	-75.83	8.23%	-63.82	11.40%	-1070.11	10.95%
2012	96.48	-1.80%	89.15	15.86%	88.31	2.64%	87.44	-0.18%
2013	92.09	8.42%	92.53	0.17%	95.56	-0.57%	96.08	5.70%
2014	95.40	3.21%	94.23	3.59%	95.30	2.29%	95.29	6.68%
2015	93.69	-1.26%	82.52	4.25%	95.52	-2.82%	94.88	-1.32%
2016	96.35	-2.89%	91.70	-2.28%	96.33	-5.20%	96.56	-6.42%
2017	76.83	-0.63%	80.44	6.51%	87.40	-3.34%	88.73	-0.45%
2018	94.09	4.19%	93.91	6.17%	97.28	3.31%	97.94	0.42%
2019	97.11	4.15%	96.13	4.31%	97.32	2.94%	97.07	2.46%
2020	97.22	5.42%	78.52	-17.76%	96.66	3.03%	97.13	5.69%
All Years	77.28	1.18%	71.73	5.37%	74.75	1.65%	56.31	2.45%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table G.3: Out of sample R^2 and MPE by year (Reduced data set, Office only)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	95.50	-0.25%	84.33	17.56%	94.86	1.66%	94.72	-7.98%
2002	0.84	12.95%	40.18	5.13%	17.32	9.73%	43.42	0.35%
2003	92.56	3.61%	75.69	7.15%	88.56	4.96%	94.72	-1.43%
2004	84.71	9.10%	57.80	17.89%	96.66	3.97%	96.06	4.94%
2005	91.20	8.66%	90.17	6.66%	89.69	8.67%	93.58	8.88%
2006	87.60	2.89%	74.70	39.43%	86.86	2.11%	86.00	17.79%
2007	66.29	3.54%	79.27	2.55%	77.26	4.49%	76.50	3.71%
2008	71.43	-10.83%	76.12	-7.25%	73.15	-4.98%	74.56	2.84%
2009	57.33	-13.18%	69.78	-8.48%	71.94	-27.42%	76.11	-1.85%
2010	83.25	10.16%	94.07	10.92%	94.77	11.78%	95.03	17.73%
2011	95.22	-17.06%	82.29	19.64%	95.21	10.68%	96.34	7.98%
2012	96.59	-6.44%	95.80	-4.52%	95.86	-0.59%	95.58	-5.89%
2013	81.38	0.46%	96.82	5.22%	96.89	8.39%	96.88	2.03%
2014	96.97	0.14%	96.04	1.34%	97.13	1.62%	96.49	-1.35%
2015	89.91	-0.79%	94.34	-1.22%	93.39	-1.79%	93.83	-2.01%
2016	88.28	-3.17%	92.48	2.30%	91.77	-1.24%	90.90	3.93%
2017	96.16	-1.23%	90.45	1.15%	96.99	-0.52%	96.77	-1.15%
2018	98.92	1.67%	98.03	-1.98%	98.63	0.59%	98.73	3.91%
2019	97.69	1.94%	96.84	3.58%	97.73	1.41%	97.95	3.37%
2020	98.70	4.22%	97.04	2.90%	96.69	6.01%	95.98	3.44%
All Years	89.39	1.41%	90.25	6.33%	92.72	2.58%	92.69	3.42%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table G.4: Out of sample R^2 and MPE by year (Reduced data set, Apartment only)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	94.72	4.00%	75.83	-0.58%	95.48	3.51%	96.38	1.22%
2002	78.98	-1.15%	90.48	2.07%	89.45	2.14%	91.41	-1.59%
2003	91.65	-0.11%	72.72	3.39%	92.31	-1.40%	93.82	0.40%
2004	82.82	4.72%	84.73	0.02%	87.31	1.91%	77.37	2.14%
2005	46.80	12.61%	49.12	13.46%	46.38	12.57%	49.26	12.32%
2006	88.93	4.54%	76.24	4.23%	89.27	2.38%	86.35	2.13%
2007	92.28	-1.55%	94.06	-3.29%	94.01	-2.02%	94.06	-1.00%
2008	86.77	-9.94%	81.68	-14.17%	89.54	-8.54%	87.6	-9.04%
2009	47.58	-17.76%	81.92	-9.56%	54.93	-7.39%	51.68	-9.54%
2010	87.60	2.57%	91.56	8.44%	91.63	4.99%	93.81	8.60%
2011	96.28	-2.20%	91.59	0.66%	95.87	-0.59%	96.52	-2.38%
2012	96.18	-3.97%	90.12	-3.47%	94.22	-3.22%	94.5	-3.48%
2013	94.75	0.47%	90.93	3.36%	94.73	3.18%	94.51	4.45%
2014	87.72	3.72%	89.61	4.39%	89.11	4.60%	88.55	3.07%
2015	82.25	5.27%	80.56	7.51%	83.29	4.38%	84.39	4.88%
2016	97.17	-0.69%	94.12	0.24%	97.52	-0.01%	97.95	0.90%
2017	95.67	-2.23%	97.55	-2.12%	98.09	-2.30%	98.02	-1.78%
2018	97.76	1.07%	93.57	0.58%	97.52	0.49%	97.45	1.21%
2019	95.36	-0.76%	93.77	1.17%	96.49	0.04%	96.12	-0.54%
2020	98.69	0.79%	98.54	0.89%	98.83	1.19%	98.74	-0.79%
All Years	88.44	0.66%	88.18	1.48%	89.05	1.28%	89.33	1.21%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table G.5: Out of sample R^2 and MPE by year (Reduced data set, Retail only)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2001	76.15	1.12%	75.40	2.37%	75.86	0.96%	78.97	18.95%
2002	93.89	6.25%	92.74	-1.80%	94.16	4.31%	92.66	2.68%
2003	85.00	3.02%	84.89	9.81%	85.95	0.50%	92.56	0.35%
2004	95.55	7.42%	86.82	5.46%	94.65	3.22%	92.11	4.82%
2005	93.85	8.13%	84.95	9.36%	92.58	8.45%	93.68	6.70%
2006	96.38	3.02%	96.35	1.57%	96.48	-0.53%	96.96	1.28%
2007	86.80	-0.94%	89.98	3.69%	89.74	4.61%	92.43	-3.17%
2008	91.78	-3.74%	71.88	-6.44%	67.75	-11.22%	74.39	-12.55%
2009	80.66	-21.12%	69.81	-18.16%	78.38	-21.46%	70.51	-22.59%
2010	91.06	-0.70%	89.18	0.51%	88.57	2.68%	90.17	3.79%
2011	92.25	14.54%	93.62	-12.25%	94.50	0.08%	93.08	16.58%
2012	90.24	-3.65%	74.17	-3.09%	92.46	0.66%	90.81	-1.68%
2013	95.39	9.43%	87.98	17.14%	95.88	14.80%	96.17	17.07%
2014	93.92	2.55%	92.32	4.74%	90.14	-0.40%	92.93	3.64%
2015	90.23	10.15%	94.78	6.07%	94.55	9.99%	93.04	4.13%
2016	97.63	2.56%	92.49	4.44%	97.11	4.95%	96.45	2.72%
2017	71.66	-7.71%	93.97	-5.83%	85.17	-8.06%	85.11	-8.73%
2018	94.30	-5.77%	98.06	-7.49%	95.68	-0.20%	95.14	-6.69%
2019	69.70	-0.60%	95.98	-0.38%	97.09	-1.11%	97.65	-1.85%
2020	97.75	-3.29%	93.98	-8.29%	95.36	-4.91%	94.81	-3.24%
All Years	83.19	3.28%	89.22	3.11%	93.31	2.88%	93.70	3.94%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.

Table G.6: Out of sample R^2 and MPE by year (Reduced data set, Hotel only)

	Model 1		*Model 2*		*Model 3*		*Model 4*	
Year	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE	R^2_{oos}	MPE
2011	-71.50	121.21%	87.11	159.13%	-228.48	480.83%	-40.21	195.00%
2012	72.12	39.89%	36.36	15.93%	68.30	47.12%	84.76	8.36%
2013	92.86	8.49%	91.89	9.90%	90.98	9.73%	94.88	9.49%
2014	94.06	3.51%	88.27	15.51%	88.24	16.06%	89.11	22.37%
2015	90.63	11.22%	87.22	12.54%	87.22	12.54%	90.60	13.02%
2016	97.71	-1.59%	95.58	-0.24%	95.58	-0.24%	94.76	-3.25%
2017	83.28	5.24%	90.06	3.96%	90.06	3.96%	92.77	0.25%
2018	74.94	0.94%	81.26	-4.77%	79.35	1.12%	93.48	-4.80%
2019	72.49	-8.82%	61.31	-7.56%	61.31	-7.56%	-71.30	-10.11%
All Years	82.97	8.79%	85.92	21.38%	83.30	27.42%	89.53	19.18%

Notes: This table reports the out-of-sample R^2s and mean percentage errors of various models. R^2s are expressed as a percentage.