# Anchoring the price Yadidadida

Philipp Gottschalk, Henry Heppe, Timothy Nijhuis, Igor Pradhan

March 2024

**FEWSIBEOR**

**Team 8**

**Word Count (abstract):**

**Word Count:**

**Abstract**

Abstract here

# 1 Introduction

With the ever-expanding presence of online marketplaces, the ability for private individuals to (re)sell goods has become ubiquitous, regardless of whether these goods are durable or non-durable. New challenges directly stem from this increased opportunity for resale. In contrast to large-scale, commercial sellers, which likely have entire teams dedicated to determining optimal listing prices for certain products, individual sellers are unlikely to always possess the level of expertise that is required to determine a listing price which is simultaneously appropriate and competitive. With professional appraisal being both costly and time-consuming, this is the problem that our paper seeks to address.

The question of determining appropriate pricing has concerned economists for decades. A popular approach to determining pricing is the hedonic pricing model, first used in literature by Court (1939). The main stipulation of this model is that the price of a good can be directly attributed to a sum of utilities derived from the attributes of the aforementioned good. This is the most appropriate model to use, within our research, in particular when considering our focus on the listing prices of boats.

There is a strong case to be made that boats are an appropriate good within the context of the hedonic pricing model. Consider that boats have isolated and distinct attributes which are likely to contribute towards the utility of the boat as a whole. Customers will have clear preferences for various characteristics of a boat. They may want a boat of a certain length or with a certain minimum engine horsepower. The synthesis of the utility that a customer receives from the fulfilment of his requirements is likely to be the most significant factor in determining the price a customer is willing to pay for the boat, in line with the assumptions of the hedonic pricing model.

Within our paper, we seek to implement and extend upon the framework first introduced within Potrawa and Tetereva (2022). This paper deals with the housing market, another common focus of hedonic pricing literature. It offers a three-step framework, consisting of feature extraction, predictive modelling using machine learning and explainable AI. We aim to apply this framework to the boat market, which, despite having a size of $35 billion in 2023 Verified Market Research (2024) has a significant scarcity of literature.

Section 2 summarises the literature concerning hedonic pricing models, machine learning applications within this framework and boat pricing. Section 3 provides an overview of our dataset and pre-processing, which was scraped from boat24. Section 4 describes

the methodology and models utilised within our paper. Within Section 5, we describe the results of our modelling. Lastly, Section 6 contains our conclusions and a discussion of our paper.

## 2 Literature Review

Court (1939) is widely considered the first piece of literature to deal with hedonic price analysis, with a focus on the automotive industry. Goodman (1998), finds that Court (1939) holds up relatively well under the standards of contemporary hedonic price analysis. Wing and Chin (2003), which aims to provide a summary of hedonic pricing literature as a whole, notes two influential papers which are usually used as the basis of modern hedonic price analysis: Lancaster (1966) and Rosen (1974).

Whilst all approaches stipulate that goods possess attributes that form bundles that the consumer values, there are key differences. Lancaster (1966) focuses on so-called consumer-theory. This suggests that all goods are members of a certain group and that combinations of these goods are consumed subject to the customers budget. In contrast, Rosen (1974) assume that a range of goods are consumed discretely. Note here that hedonic pricing, similar to the theory in Rosen (1974) does not require goods to be consumed jointly. In our paper, which is focused on boat data, Rosen (1974) is the most appropriate basis, as boats are highly unlikely to be subject to any form of combined consumption except for the richest of customers.

Conventionally, the standard approach to hedonic pricing is the use of OLS-based models, as noted by Potrawa and Tetereva (2022), with machine-learning models still limited in the literature, especially due to their black-box nature and subsequent lack of interpretability. As noted above, Potrawa and Tetereva (2022) enhance the hedonic pricing model by proposing a general framework for using machine-learning tools. This framework consists of *Feature Extraction*, *Predictive Modelling* and *Explainable AI*, which focuses on the interpretation of results. With regards to *Feature Extraction*, notable features that Potrawa and Tetereva (2022) use are extracted from images from the rental offers, text data from the descriptions of the apartments, as well as geographical data. The geographical data include the latitude and longitude of the house offered, as well as the distance to the central business district. Latitude and longitude can be used as

proxies to seperate houses into neighbourhoods. For *Predictive Modelling*, Potrawa and Tetereva (2022) use decisions trees in the form of random forests and an OLS model as a comparison, finding that random forests have a lower prediction error. Lastly, in the *Explainable AI* part of the framework, Potrawa and Tetereva (2022) use three model agnostic methods, to remove the "black-box" aspect of the model: variable importance, partial dependence analysis and local interpretable model-agnostic explanations (LIME).

There are other papers which use machine-learning in an attempt to improve traditional hedonic pricing methodology. Hong et al. (2020) use random forests as a predictor for pricing, rather than a traditional hedonic pricing model, finding that random forests may serve as an appropriate complement to hedonic pricing models, as they more adequately model non-linearities in the relation. Chen et al. (2020) not only use machine-learning models for prediction, but also introduce Shapley values to better interpret the machine-learning. Similar to Potrawa and Tetereva (2022), it also uses a framework with machine learning in all three aspects: feature extraction, predictive modelling and model interpretation.

A study that incorporates a large number of machine-learning models is Garcia (2023), which looks at the effect of introducing image features to machine-learning methods in a hedonic pricing context. During the investigation, multiple machine-learning models are tested, whilst measures such as H-statistics or Shapley values are used for interpretability.

The literature specifically concerning the hedonic pricing of boats seems to be limited at best. Akyurek (2013) looks at a hedonic pricing model of 'mega yachts', finding that the length of the yacht seems to have the largest effect on the yacht's price, in the sense that a percentage change in length corresponds to the largest percentage increase in price. A paper that explores the issue of boat pricing, although not utilising a hedonic pricing model, is Chen et al. (2024), which carries out sailing boat pricing using Principal Component Analysis and Back-propagating Neural Networks. However, this study is limited to sailing boats alone.

To see the relevance and impact of the boat market, one could look at the literature concerning boats as a luxury consumption good. Note that, to the best of our knowledge, this literature is severely limited. An example of the relevance of the boat market is that, as stated in Amatulli et al. (2017), despite the (at the time) recent global recession, the pleasure boating sector was among the sectors with the highest capacity to generate

wealth and employment.

# 3 Data

In this section the collection of boat pricing data is described and an overview over the data is provided. For the extraction of features from the unstructured data refer to Section 4.1.

All of the data used here were collected from the online boat marketplace boat24. Such websites function as a modern day catalog for boats. Potential sellers can upload a sales advertisement of their boat and potential buyers use the platform to find candidate boats or simply explore the market. A transaction is not completed via the platform, its use is mainly to connect sellers and buyers (similar to real-estate online marketplaces). The advertisements provide information on the boat characteristics, images and location of the boat and a price. This price will usually differ from the price a boat is sold for as it is only a first offer. However, sales prices are not publicly available and listing prices provide a reasonable proxy. This is because the focus of the analysis is to give guidance to potential new sellers on how their boat could be priced. Now, whether this guidance is in terms of the true price or listing price is not of great importance as such a seller can make use of the information accordingly. The listing prices additionally have the advantage that anyone can do a sanity check on the model prediction by looking up listing prices of similar boats.

The data were collected in March 2024 and provide a snapshot of this part of the boating market. Per boat advertisement the price, a large set of boat characteristics, the boat location (on country level) and images of the boat were collected. This mimics (similar to Obaid et al. (2019)) the intuitive data collection behaviour of a potential buyer. After scraping the website and removing everything that is not a boat, does not have a price, is a new boat or is a duplicate we are left with 14439 observations. All boats with a price over £500,000 were also removed as the number of data points above is too low to reliably predict prices for those boats. A histogram of prices in the data and an example image are shown in Figure 1.

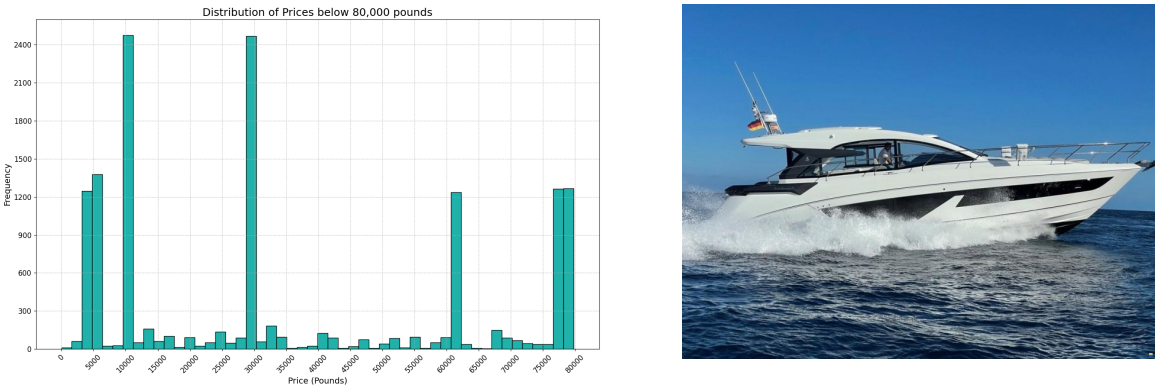


Figure 1: Distribution of prices for boats below £500,000 and an example image of a boat on sale

For the image collection the first 3 images per boat were taken into account. The number of images per boat ranged from 0 to over 100 but only few boats had less than 3 images. This decision is supported by the fact that when opening an advertisement on the website one only sees the first 3 images directly. They can thus be conjectured to have the largest impact on the viewers impression of the boat. This is augmented by the fact that scraping and classifying one image per boat takes around 2 hours. This resulted in about 28,000 images that were collected. For the image classification model a training set of 1000 images were manually labeled with an initial validation set of 52 images. To check the reliability of the model an extended test set of another 200 images was created.

The geographical data are included since their relevance in other hedonic pricing studies (see Section 2) can be an indication for this application as well and this type of data can provide a useful geospatial interpretation of the machine learning models. For example the decision trees in Random Forests partition the feature space into regions and thus also effectively partition the geographical feature space. This partition can then be interpreted as neighborhoods which make for a useful proxy variable for a set of unobserved variables that are suspected to have great influence on the price (in housing this could be for example air quality and neighborhood crime rates). Making use of this proxy nature of geographical data, one can make a similar case for including it when analyzing boat sales. For example, local market dynamics could play a non-negligible role for the price. As such the demand and willingness to pay for a super yacht sold in St. Tropez could be much higher than in a small town in Norway. Additionally, not only the type of customers would differ greatly between those locations but also the type of boat demand induced

by the environmental conditions. Although boats are certainly not bound to their sale location in the way a house is, transportation of a boat can be costly which could be taken into account for the price.

Since a large set of heterogeneous boats are sold on the marketplace, only a subset of collected variables apply to a given boat resulting in many missing values. It is not an option to remove all observations that are incomplete, as no observations would be left. In this paper two different imputation approaches are chosen: semi-manual and simple imputation (abbreviated as datasets *man* and *freq*).

The semi-manual approach uses domain knowledge to impute values where possible. For example missing values in variables representing sail area are set to zero for boats that are not sailing boats. For the remaining missing values imputation is done by estimating the relationship between the variable in question and related variables with a Random Forest. This is done with the **?** R package where the idea is to treat the imputation problem as a supervised learning problem. The Random Forest takes the variable in question as a dependent variable and all other given variables as independent variables. By estimating this interplay on the non-missing values it can then predict a missing value based on the variable values of the other variables in the same row. For imputing one variable only a subset of all other variables is taken into account because of computation time limits. This subset is determined based on domain knowledge as well. For example draught and displacement can be hypothesized to be imputed well based on each other and length and beam of the vessel.

The simple approach is called *freq* because it imputes every missing value with the value occurring most frequently for the respective variable (i.e. the mode of the column).

Both approaches are compared in the results section to provide an idea whether it is worth putting in the effort to use domain knowledge for a more careful imputation of whether a simple imputation approach suffices for such data. Additionally, previous research (see Le Morvan et al. (2021)) suggests that a powerful learner like a Random Forest can also yield asymptotically optimal predictions when the data are imputed with a simple imputation mechanism while that is not the case for weak learners.

# 4 Methodology

Since our research is based on the three-step hedonic pricing framework used by Potrawa and Tetereva (2022) the methodology is split into feature extraction, price modelling and interpretation as well.

## 4.1 Feature extraction

In this section we elaborate on how exactly the data (described in Section 3) enter the different models. First we explain which features we extracted from the images followed by the transformations of the other variables together with overview tables of our data.

### 4.1.1 Image Classification

The approach taken in this paper is that of extracting pre-specified features of the images. We do not include the images themselves in for example a neural network to directly use them for predicting prices as that would not allow us to easily understand the importance and effect of different characteristics of the image.

From the images three features per boat are extracted. First a binary variable *image water* which is 1 if among the three images at least one of them shows the boat on the water. Similarly, the second variable *image land* is 1 if among the three images at least one of them shows the boat on land. These two variables could be relevant since a boat on water might be perceived as more appealing while a boat on land usually shows the underwater body of the boat - the condition of which is usually a highly relevant quality attribute.

The third feature is *image brightness* with respect to the first image. Image brightness could be conjectured to be relevant in that brighter images of boats are to a degree perceived as more aesthetically pleasing drawing in the attention of the user. Brightness is also associated with good weather in the image which certainly paints the boat in a better light. One can argue that these effects are mainly relevant for the first impression of the boat, which is why we only measure it for the first image. This first image is shown on the marketplace as a thumbnail, i.e. when scrolling through the list of boats it is only the first image one sees. Hence, it plays the largest role among all the images in generating demand for the boat through affecting the click-through-rate.

Although *image water* and *image brightness* are characteristics of the advertisement and not of the boat itself these features are included in the analysis because of possible priming effects on the customer's willingness to pay and to be able to give advice to sellers on whether the quality of the advertisement is also important to their sale instead of only the boat itself.

For classifying whether an image shows the boat on water or on land we finetune a deep learning image classification model on our training dataset. We use the *ConvNextV2* model architecture by Woo et al. (2023). This architecture is an update to the ConvNext Liu et al. (2022) version of convolutional neural networks. This update introduces a self-supervised learning component to the originally purely superincludes a fully convolutional masked autoencoder framework and a

For the water/land classification we finetuned the ConvNextV2-tiny model on our training dataset of 1000 labelled images. After a training time of only a few minutes, the model achieved a prediction accuracy of 91.8%, making it highly effective for this classification task. We also finetuned a Vision Transformer (Dosovitskiy et al. (2020)) and a DINOv2 model (Oquab et al. (2023)) both of which required much longer training times but were not able to improve their predictions beyond 85% out-of-sample accuracy. Taking into account the disbalance in our training dataset (only 17.6% of images show a boat on land) the goal was for the model to make predictions that are better than the approximately 82.4% accuracy one would get by always predicting water.

The features are extracted by first obtaining an abstract representation of the images from the recent Image Joint-Embedding Predictive Architecture (I-JEPA by Assran et al. (2023)). The paper introducing this model suggests that the image representations obtained from this model can be used for applications such as image classification requiring less compute and less training data than previous state-of-the art approaches. Since this model does not require fine-tuning to provide useful representations and our research is constrained in terms of available compute and labelled data, this model is used to extract the features.

### 4.1.2 Numerical and Categorical Data

In total a set of 69 features is recorded in the imputed datasets. Among them are numerical features such as *Age, Length, Beam* and *Total Horsepower*, binary features like *Motorized*

*Vessel*, unordered categorical features and the image-based and location-based features. Relevant summary statistics and an example observation are included in Table 2 showing some of the most important characteristics of a boat. A full list of the variables can be found in the Appendix.

| | Price (£) | Age (years) | Condition | Length (m) | Beam (m) | Certified No. of Persons | Engine Hours |
|---|---|---|---|---|---|---|---|
| Example | 583100 | 2 | 7 | 14.78 | 4.2 | 12 | 265 |
| Mean | 107788.2 | 18.1 | 5.6 | 9.6 | 3.1 | 8.4 | 908.7 |
| StD | 109913.9 | 18.7 | 0.8 | 3.7 | 1.1 | 3 | 1206.3 |
| Max | 499200.0 | -1.0 | 8.0 | 50.0 | 55.0 | 26 | 1.0 |
| Min | 1.0 | 157.0 | 1.0 | 1.0 | 0.3 | 1 | 20500.0 |
| NA | 0 | 0 | 8195 | 0 | 0 | 7137 | 7476 |

Table 1: Example boat characteristics and summary statistics for a subset of variables of the collected and cleaned boat pricing data.

For some of the other variables it is worth elaborating on what they mean and where they come from. *Condition* is an ordered categorical variable taking values from 1 to 8 which was created based on the conditions specified in the dataset. *Ship area* is created from multiplying length times beam. *Negotiation possible* summarizes as a binary variable a set of different spots on the advertisement where one can mention that the listing price is explicitly up for negotiation. Similarly *Taxes Paid* simplifies a whole set of different tax rates that are often marked as paid. *CE Design Category, Hull Type, Material, Steering* and *Keel Type* are all unordered categorical variables that are encoded as binary dummy variables for each category. *Big equipment index* and *Small equipment index* were created as a variable summarizing the equipment items coming with the boat. Based on the value of the equipment we divided them into two groups and counted the number of equipment items in each group per boat. This was done because there is a large number of possible items where each boat is equipped with only a small number of them. The summary variables are used to avoid burdening the models with many more highly sparse columns.

The country variable is translated into two features *Latitude* and *Longitude*. They point to a representative point in the country where the boat is sold. For the sake of simplicity only the country is used as a location, but they are still converted to coordinates to make use of the geospatial modelling capabilities of the machine learning models. With the coordinates the model can take into account distances and implicit similarities between countries, which would not be the case if the countries were included as a purely categorical variable.

The two imputed datasets *man* and *freq* show two characteristics that can pose a problem to some of the machine learning models. They have a large number of columns, some of which are highly sparse and/or correlated and some of the variables suffer from perfect multicollinearity. We perform two transformations of the variable space to deal with these problems.

The first transformation of the feature space is done by applying Principal Component Analysis to the variables *Certified No. of Persons, Cabins, Berths, Bathrooms, Toilets* and *Showers* as they are highly correlated (see Figure ?? in the Appendix) and replace them with the first component. Similarly we reduce *Mainsail Area, Jib Area, Genoa Area* and *Spinnaker Area* to one component. For the *man* data the first components explain more than 75% and 85% of the respective variance and for the *freq* data they cover more than 60% and 75% and of the variance respectively, justifying this reduction. This yields the datasets *man_PCA* and *freq_PCA*. The idea of this approach is to see if it improved the performance of models which have problems with the high dimensionality of the dataset such as the neural network for which the ratio of sample size and number of variable is at the lower end of the acceptable.

The other data transformation is aimed at the linear and penalized regression models. For the nonparametric models the dummy encoding for categorical variables is done by including one variable per category. However, this means that for $k$ categories this means that the $k-th$ dummy variable is perfectly predicted by the $k-1$ other dummy variables, which results in perfect multicollinearity implying the models cannot be estimated. We therefore remove one dummy variable per original categorical variable which then appears in the model as a baseline category. This yields the two additional datasets *man_BaseCat* and *freq_BaseCat*.

In total we thus have six datasets which we run our experiments on: the two purely imputed datasets, the two PCA datasets and the two base category datasets.

## 4.2 Predictive Modelling

This section is dedicated to summarizing the main characteristics of the different models that are employed for predicting a boat's price based on its characteristics. A wide range of models in covered on purpose, as for this step of the framework the goal is to achieve the best prediction accuracy. The interpretability of the following models varies greatly,

also depending on the exact hyperparameters. That is why for all of the methods the explainable AI methods of the third part of the framework by Potrawa and Tetereva (2022) are applied whatsoever.

At first support vector machines and more complex Gaussian process regressions were also considered for modelling the price-attribute relationship but were discarded later on due to a mismatch in their theoretical strengths and the application here. Both models are not commonly used for regression tasks where the relationship between predictor and dependent variable is assumed to be nonlinear (see Williams and Rasmussen (2006)).

### 4.2.1 Linear Regression Model

First a standard linear regression model estimated with ordinary least squares is applied to the data. This has the advantage that it sets a benchmark on predictive performance against which the advanced nonparametric methods can be compared. All variables are included as regressors initially in their original form. As a second step standard transformations such as semi-log and Box-Cox transformations are used on the dependent variable to bring the data closer to the assumptions underlying the model if possible. Moreover, using a power transform on the regressors helps reduce the variance, and make the data more normally distributed Yeo and Johnson (2000). Possible misspecifications, heteroskedasticity, endogeneity of regressors and other problems are not further investigated, as this serves solely as a benchmark. A degree of interpretability similar to this model is reached for the other models by applying the explainability techniques of Section 4.3.

### 4.2.2 Penalized Regression

An easy way to improve on the predictive performance of the standard linear regression model is to use penalized regression models (see James et al. (2013)). To avoid overfitting when including all features in the regression, these models regularize the coefficient estimates. That is one introduces a penalty term which shrinks the regression coefficients towards zero. Here, we consider Ridge regression, Lasso and their combination the Elastic Net regressions. Ridge regression minimizes the standard residual sum of squares loss

from OLS with the additional penalty term, namely the $L_2$ penalty

$$\lambda \sum_{j=1}^{p} \beta_j^2,$$

for regressors $j = 1, \ldots, p$, where $\lambda$ is a hyperparameter tuned via cross validation. Combining the penalty term with the original loss function of a linear regression gives the following result:

$$\frac{1}{2n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

For the Lasso, the penalty term is the $L_1$ penalty

$$\lambda \sum_{j=1}^{p} |\beta_j|.$$

Similarly to the ridge regression this penalty term is coupled with the OLS loss function

$$\frac{1}{2n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Finally we also consider the Elastic net. Elastic net includes both the $L_1$ and $L_2$ penalties at the same time each with their separate hyperparameter $\lambda_1$ or $\lambda_2$. We then mimise the following objective:

$$\frac{1}{2n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2$$

### 4.2.3 Optimal Sparse Regression Trees

Optimal Sparse Regression Trees (ORSTs) are a method introduced by Zhang et al. (2023b). This approach builds on the model first introduced by Bertsimas et al. (2017), which suggests solving an entire decision tree in one go to ensure global optimality. This is achieved by solving a Mixed-Integer Optimisation (MIO) problem rather than using greedy heuristics, as is conventional for regression trees, described in James et al. (2013).

Zhang et al. (2023b) proposes a dynamic-programming-with-bounds approach to construct provably-optimal sparse regression trees. To do so, they create a lower bound based on an optimal solution to the k-Means clustering algorithm. Zhang et al. (2023b) claims that this creates an OSRT which creates a fast, consistent and interpretable model.

Denoting the mean squared error (MSE) as $\mathcal{L}(t, \mathbf{X}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$, where $\hat{y}_i$ is

the prediction of $x_i$ by tree $t$, Zhang et al. (2023b) denotes the objective function as:

$$\mathcal{R}(t, \mathbf{X}, \mathbf{y}) = \mathcal{L}(t, \mathbf{X}, \mathbf{y}) + \lambda H_t$$

Here, $H_t$ is the number of leaves in tree $t$, whilst $\lambda$ is the hyper-parameter which punishes model complexity. Note that adding a depth constraint is possible, but also introduces the possibility of sub-optimal values of the objective, should the depth of the optimal solution be higher than the depth constraint.

Note also that in Zhang et al. (2023b), the data is binarised, to allow for timely computation of the OSRT. Within our application of the OSRT, we do the same, by using the guessing_threshold function available in Zhang et al. (2023a), the Python package associated. This approach is introduced in McTavish et al. (2022). Note that this can impact optimality, but this is necessary for the OSRT to be computationally feasible.

Further, we set the similar_support setting of the OSRT configurations to True (for a complete list of the configuration, see the Appendix). This allows the OSRT to use similar support bounds in the creation of the tree. As described in Lin et al. (2022), if two features in the dataset are similar, bounds obtained for one feature to create a split in the tree can be leveraged to create bounds for the second feature in the tree, were the second feature to replace the first one. This allows us to further reduce the search space.

#### 4.2.4 Random Forests

Random Forests are a solution to the high variance problem of single regression trees (see James et al. (2013)). They reduce the variance by estimating a large number of small trees (so called weak learners) and averaging the predictions over all trees. The estimated trees differ from one another in that they are all trained on a different random subsample of the original data (also called bagging). To decorrelate these trees from one another a tree considers only a subset of the features when estimating a new split. That results in dominant features occasionally not being considered at every step in a tree, hence the decorrelation. Random Forests have showed strong predictive power in practice, however, they lack the inherent interpretability of single trees.

### 4.2.5 Gradient Boosting Trees

Some of the more recent methods with the strongest prediction accuracy track record over a wide range of tasks haven been methods based on gradient boosting. Boosting is another ensemble method that estimates many regression trees to get better out-of-sample prediction accuracy (see James et al. (2013)). Here the trees are not estimated separately, but sequentially in a way that each tree uses information from the previously grown trees by fitting the next tree to the previous tree's residuals. The term gradient boosting stems from the view on the sequential fitting as a gradient descent algorithm. Estimating these models gives rise to three hyperparameters: number of trees $B$, shrinkage parameter $\lambda$, number of splits $d$ in each tree. The framework implementing gradient boosting in an open source python package used here is LightGBM (Ke et al. (2017)).

### 4.2.6 Bayesian Additive Regression Trees

One last ensemble tree method to be evaluated on boat price predictions is the Bayesian Additive Regression Tree (BART) model. James et al. (2013) describe them as combining the approaches of boosting and bagging. In this approach broadly speaking trees are fitted sequentially on a previous tree's (partial) residuals but at the same time a perturbation to that previous tree is chosen randomly, where perturbations improving the partial residual fit are more likely. The hyperparameters to tune in cross validation are the number of trees to estimate, the number of iterations and the number of burn-in iterations (which are needed because the first iterations are usually not useful in this model).

### 4.2.7 Neural Network

## 4.3 Model Interpretation (Explainable AI)

After identifiying the best performing model among the different machine learning approaches, that best model is used for a more precise analysis and interpretation of how it comes to its predictions and what role the different boat attributes play. This is central to our goal of providing potential boat sellers with an idea for which characteristics of their boat drive the price the most, i.e. in which part of the boat it might be worth investing some more money before selling it.

### 4.3.1 Variable Importance (LOCO)

A central question when interpreting more complex models is which variables drive the predictive power of the model. Also called variable importance, this is measured by examining how the prediction accuracy of the model changes when one variable is left out of the model. This Leave-One-Covariate-Out (LOCO) approach (by Lei et al. (2018)) has the advantage that it is model agnostic. The measure can be relative $VI_j = e_{-j}/e_{full}$ or absolute $VI_j = e_{-j} - e_{full}$ where $VI_j$ is the variable importance score of variable $j$ and $e$ is the out-of-sample prediction error. In this study the relative score is used. We chose the LOCO approach over the more commonly seen Shapley values, as it is deemed to better address correlation between the variables (see Verdinelli and Wasserman (2023)).

### 4.3.2 Partial Dependence Analysis

A graphical approach to interpretability we use is the analysis of partial dependence plots (PDP). They provide an easy-to-understand way to visualize the effect of one variable on the outcome variable after the effects of all other variables were accounted for. The plots can be obtained by averaging the prediction function over all values found for the other variables in the dataset. One can then quickly see if the effect is somewhat linear, piecewise linear or nonlinear. Hastie et al. (2009) emphasize that this approximation of the effect is different from the partial effect interpretation you would have in a simple regression model where you ignore all other variables (which would be the conditional expectation and only equal to the approximation if the variable in question is independent from all others).

### 4.3.3 Local Interpretable Model-agnostic Explanations (LIME)

Previously described interpretation methods are focused on global interpretability. To explain individual observations, we utilise the Local Interpretable Model-Agnostic Explanations (LIME), introduced by Ribeiro et al. (2016). Within this investigation, we implement LIME using the associated Python package, Ribeiro (2019).

The intuition behind this model is as follows: in any non-linear model, observations which have extremely similar covariates should produce similar outcomes. Hence, there should be a small enough local area which can be approximated as a linear model.

This approach is explained in a relatively simple manner in both Molnar (2019) and

Potrawa and Tetereva (2022). First, we choose the observation for which we wish to explain the black-box prediction. We perturb the dataset and find the predictions for the new data points. The package by Ribeiro (2019) uses Gaussian sampling as a default. We then weigh the new samples based on the distance between the original observation and the predictions for the perturbed data. We then fit an interpretable linear model on this weighted data set. In our paper, we choose to use an Elastic Net regression, as described in Section 4.2.2, combining the benefits of LASSO and Ridge Regressions.

Examples of the use of LIME to explain prediction values of individual observations can be found within Section 5. We also discuss the exact interpretation of a LIME model.

# 5 Results

# 6 Conclusion

# 7 Methodology Timothy

## 7.1 Ensemble Models

This paper treats three different ensemble models. For ensemble models the prediction is given by.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathcal{F} \tag{1}$$

The objective function for ensemble models contains (i) a loss term and (ii) a regularization term. This can be represented as.

There are two common methodologies. The first is an ensemble bagging algorithm, as used in a Random Forest model. This algorithm creates a large number of independent trees that are de-correlated and then averages over the tree predictions for a final prediction. The gradient boosting algorithm however adds trees sequentially, where each tree is meant to correct for the predictions by the previous trees. This algorithm uses gradient descent to determine the structure of the new tree that will be added.

$$\text{obj} = \sum_{i=1}^{N} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \omega(f_k) \tag{2}$$

To train trees an additive approach is used. This can be represented by the recursion formula:

### 7.1.1 XG UltraBoost

The loss part

Since we use the reg:squarederror parameter in our XGBoost implementation, we can simplify the notation of the loss part to

$$\sum_{i=1}^{n} \left[ 2 \left( \hat{y}_i^{(t-1)} - y_i \right) f_t(x_i) + f_t(x_i)^2 \right] + \omega(f_t) \tag{3}$$

Note that due to the additive training approach of trees, we drop the constant term from the loss function. The general notation when optimizing other loss functions is given by.

$$\sum_{i=1}^{n} \left[ g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \omega(f_t) \tag{4}$$

Where g is the gradiant and h is the hessian. The regularization part

$$\omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \tag{5}$$

The first part of the regularization term penalizes the number of leaves directly. The second part of the regularization term penalizes the number of leaves indirectly, and penalizes large scores on any one leave directly.

Now with our new definition of ft we can write the complete objective function for tree t as

$$\text{obj}^{(t)} \approx \sum_{i=1}^{n} \left[ g_i w_{q(x_i)} + \frac{1}{2} h_i w_{q(x_i)}^2 \right] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \tag{6}$$

$$= \sum_{j=1}^{T} \left[ \left( \sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \sum_{i \in I_j} (h_i + \lambda) w_j^2 + \right] \gamma T \tag{7}$$

Ideally, we would enumerate all trees and pick the best one. However, this is in-

tractable. Therefore XGBoost optimizes one level of a tree at a time. It splits a leave into two and calculates the gain as

$$\text{Gain} = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \tag{8}$$

If the gain is larger than the penalty we split the leave into two.

The model is implemented with the xgboost package and the optimal parameters where found using the optuna optimization package based on 100 trial runs with 5-fold cross validation.

The parameters to set are objective ='reg:squarederror', colsample_bytree = 0.85, learning_rate = 0.096, max_depth = 8, alpha = 8.31, n_estimators = 164

(current information taken from dcml xgboost site)

## 7.2 Rambunctious Forest

The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set $X = x_1, \ldots, x_n$ with responses $Y = y_1, \ldots, y_n$, bagging repeatedly (B times) selects a random sample with replacement of the training set and fits trees to these samples:

For $b = 1, \ldots, B$:

1. Sample, with replacement, $n$ training examples from $X, Y$; call these $X_b, Y_b$.

2. Train a classification or regression tree $f_b$ on $X_b, Y_b$.

After training, predictions for unseen samples $x'$ can be made by averaging the predictions from all the individual regression trees on $x'$:

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(x')$$

Random forests also include another type of bagging scheme: they use a modified tree learning algorithm that selects, at each candidate split in the learning process, a random subset of the features

The reason for this approach is precisely to reduce the correlation between trees in the forest. If all trees were to consider all features at every split, the strongest predictors

would be used in the same manner across many trees, making them correlated (or similar to each other). By forcing each split to consider only a subset of features, it increases the diversity among the trees in the forest. Diverse trees make the forest more robust and improve its ability to generalize because they are likely to make different kinds of errors, which cancel out when aggregating their predictions.

The model is implemented with the sklearn package, which includes a Random Forest model.

The parameters to set are objective ='reg:squarederror', n_estimators=200, max_features = not done yet, max_leaf_nodes = not done yet

(current information taken from random forest wikipedia)

Suggestions for Future Research: Enriched Random Forest (ERF): Use weighted random sampling instead of simple random sampling at each node of each tree, giving greater weight to features that appear to be more informative.[18][19] Tree Weighted Random Forest (TWRF): Weight trees so that trees exhibiting better accuracy are assigned higher weights.[20][21]

## 7.3 LG BroMo

This is practically the same as the XGBoost model. However, in computation it is optimized. Moreover

LightGBM does not grow a tree level-wise — row by row — as most other implementations do.[9] Instead it grows trees leaf-wise. It chooses the leaf it believes will yield the largest decrease in loss.

The leaf-wise strategy can result in trees that are less balanced but more aligned towards reducing the overall loss. Because this method focuses on the most promising areas of the tree (in terms of loss reduction), it can often achieve better performance with faster computation, especially on datasets with large features and complex structures. However, without careful control, this aggressive focusing on the areas of greatest loss reduction can also lead to overfitting, especially with small datasets, because the model might start to "chase" outliers or noise in the data rather than learning the general pattern.

the parameters to optimize are (boosting_type='gbdt', num_leaves= 256, colsample_bytree = 0.85, max_depth= 8, learning_rate= 0.096, n_estimators= 164, subsam-

ple_for_bin=20000, reg_alpha = 8.31, random_state=randomSeed

(current information taken from LGBM wikipedia)

### 7.4 Baddie Alert Regression Trees

ski bi da bo bap

random forests and boosting models use the same models (tree ensembles), the difference arises in training

for ensemble models, the prediction is given by

## References


Akyurek, E. (2013). Pricing of mega yachts. Master's thesis, Sosyal Bilimler Enstitüsü.

Amatulli, C., Nataraajan, R., Capestro, M., Carvignese, M., and Guido, G. (2017). "service" in luxury retailing in the twenty-first century: An exploratory look at the pleasure boating sector. *Psychology & Marketing*, 34(5):569–579.

Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., and Ballas, N. (2023). Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629.

Bertsimas, D., Dunn, J., and Paschalidis, A. (2017). Regression and classification using optimal decision trees. In *2017 IEEE MIT Undergraduate Research Technology Conference (URTC)*, pages 1–4.

Chen, L., Yao, X., Liu, Y., Zhu, Y., Chen, W., Zhao, X., and Chi, T. (2020). Measuring impacts of urban environmental elements on housing prices based on multisource data—a case study of shanghai, china. *ISPRS International Journal of Geo-Information*, 9(2).

Chen, Y., Li, Z., and Jia, D. (2024). The sailing boat price study based on principal component regression analysis. *Highlights in Business, Economics and Management*, 25:189–196.

Court, A. (1939). *Hedonic price indexes with automotive examples.*

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is

worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Garcia, S. G. (2023). Evaluating the impact of image features on airbnb price predictions: A machine learning approach to hedonic pricing. Master's thesis.

Goodman, A. C. (1998). Andrew court and the invention of hedonic price analysis. *Journal of Urban Economics*, 44(2):291–298.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Hong, J., Choi, H. H., and Kim, W.-S. (2020). A house price valuation based on the random forest approach: the mass appraisal of residential property in south korea. *International Journal of Strategic Property Management*, 24:1–13.

James, G., Witten, D., Hastie, T., Tibshirani, R., et al. (2013). *An introduction to statistical learning*, volume 112. Springer.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

Lancaster, K. J. (1966). A new approach to consumer theory. *Journal of Political Economy*, 74.

Le Morvan, M., Josse, J., Scornet, E., and Varoquaux, G. (2021). What'sa good imputation to predict with missing values? *Advances in Neural Information Processing Systems*, 34:11530–11540.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., and Wasserman, L. (2018). Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111.

Lin, J., Zhong, C., Hu, D., Rudin, C., and Seltzer, M. (2022). Generalized and scalable optimal sparse decision trees.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. (2022). A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986.

McTavish, H., Zhong, C., Achermann, R., Karimalis, I., Chen, J., Rudin, C., and Seltzer, M. (2022). Fast sparse decision tree optimization via reference ensembles.

Molnar, C. (2019). Interpretable machine learning-a guide for making black box models

explain-443 able.

Obaid, H. S., Dheyab, S. A., and Sabry, S. S. (2019). The impact of data pre-processing techniques and dimensionality reduction on the accuracy of machine learning. In *2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)*, pages 279–283.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. (2023). Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Potrawa, T. and Tetereva, A. (2022). How much is the view from the window worth?: Machine learning-driven hedonic pricing model of the real estate market. *Journal of Business Research*, 144:50–65. Publisher Copyright: © 2022 The Authors.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Ribeiro, M. T. C. (2019). Lime python package. https://github.com/marcotcr/lime.

Rosen, S. (1974). Hedonic prices and implicit markets: Product differentiation in pure competition. *Journal of Political Economy*, 82(1):34–55.

Verdinelli, I. and Wasserman, L. (2023). Feature importance: A closer look at shapley values and loco. *arXiv preprint arXiv:2303.05981*.

Verified Market Research (2024). Global boat market size by propulsion system, by material, by end-user, by geographic scope and forecast. https://www.verifiedmarketresearch.com/product/boat-market/.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Wing, C. K. and Chin, T. (2003). A critical review of literature on the hedonic price model. *International Journal for Housing Science and Its Applications*, 27:145–165.

Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I. S., and Xie, S. (2023). Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16133–16142.

Yeo, I.-K. and Johnson, R. A. (2000). A new family of power transformations to improve normality or symmetry. *Biometrika*, 87(4):954–959.

Zhang, R., Lin, J., and Zhong, C. (2023a). optimal-sparse-regression-tree-public. https://github.com/ruizhang1996/optimal-sparse-regression-tree-public.

Zhang, R., Xin, R., Seltzer, M., and Rudin, C. (2023b). Optimal sparse regression trees. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11270–11279.

# Appendix

## A Addressing multicolinearity and hyperparametertuning of regressions

Considering Dataset $\{2, 3, 5, 6\}$ contain the one-hot-encoded categories without removing the base category. Without removing this we get multicolinearity and finding the inverse would be impossible. However, when using the packages SKLearn for Ridge, Lasso and Elastic net and StatsModels for OLS, it addresses the issue by taking the Moore Penrose Pseudoinverse. This adresses the issue of finding the inverse but will lead to worse results.