Prediction of Abalone rings as an approximation of age from physical measurements using Machine Learning algorithms and ensembles

Víctor Hugo Peña-García

Introduction

Abalones are mollusks of warm seas belonging to family Haliotidae which have a unique gender, *Haliotis*. Its meat is highly appreciated in some countries from Eastern Asia and recently in United States and Mexico. Its consumption has led to some problems related to its conservation, which had made the knowledge of its biology an important topic in the conservation-related sciences. In biology- and conservation-related studies, the assessment of the age is a challenging task that require training, expertise and resources, besides of time. This activity comprises the fine cutting of the shell through the cone, a staining process which involves the etching the shell sections with weak hydrochloric acid and then applying a stain to the surface. Finally, by using a microscope, the number of rings observed in this section is used as an approximation of the age of the abalone. However, other measures can be easily obtained with no major consumption of time, resources or materials like the weight, height, length, diameter, among others. It has been theorized that such measures preserves some relationship with the age so it can be predicted from these. In this work, it is reported a way to predict the amount of rings that can be used to further estimate the age of the abalone by using an ensemble of predictions done by machine learning models-choosen algorithms.

Methods and analysis

The `abalone` dataset

The dataset used for this work includes 4177 instances of 9 variables. One of the variables is the outcome `rings`, the remaining 8 are the predictors. A description of the dataset can be found at table 1.

Table 1: Variables of dataset and their characteristics

Name	Data_type	Measurement_unit	Description
sex	categorical	none	Male (M), Female (F) and Infant (I)
length	continuous	mm	Longest shell measurement
diameter	continuous	mm	Perpendicular to length
height	continuous	mm	With muscle in shell
whole_weight	continuous	grams	Whole abalone
shucked_weight	continuous	grams	Only the muscle
viscera_weight	continuous	grams	Gut weight (after bleeding)
shell_weight	continuous	grams	After dried
rings	integer	none	+1.5 gives the age in years

Pre-processing

The downloaded dataset required minor pre-processing after being obtained from the repository(UCI Machine Learning). All variables displayed some degree of significant standard deviation. However, a pair of outliers were identified that potentially could bias predictions, as can be seen in the figure 1. To solve the presence of such outliers, these were fill with mean of height which is thought as a less noisy value.

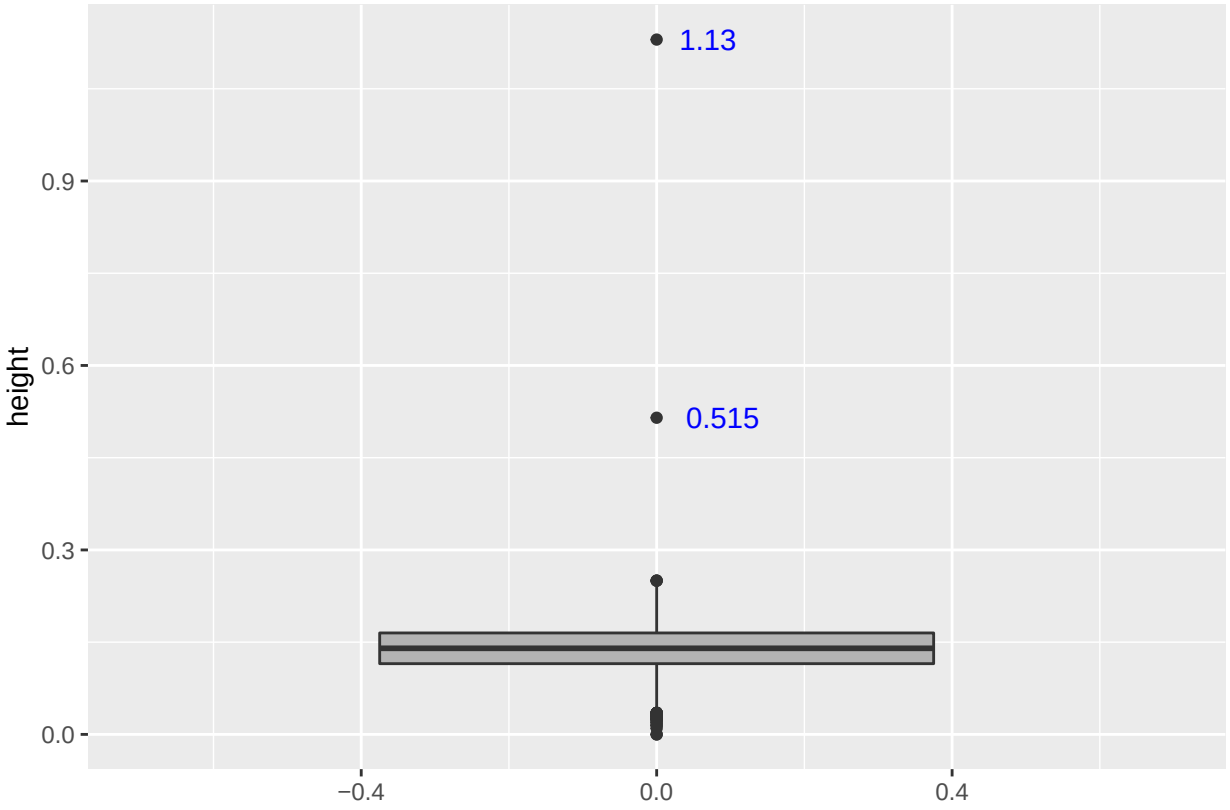

Figure 1: Distribution of variable Height. Outliers are highlighted with their respective value

To train the model and test it, the entire dataset was splitted into a validation set and train set. Since the entire dataset comprises 4177 records (not as large as, for instance, more than 10,000 data), it was desirable to ensure a good amount of data to get accurate metric of evaluation but also, enough number of data to train the model. For this reason, the dataset was splitted to get a validation test with 20% of data and a training set with 80% of data. The train set was further splitted into train set and test set to get momentarily metrics during the training process. The same rationing was applied to this split so the test set comprised 20% of data and the train set the remaining 80%.

Training the model

The response variable was the number of rings of the abalones, which can be treated as categorical or integer variable. However, some values had a frequency of 1 in the dataset, as can be seen in the figure 2, which can be problematic during the training process. Also, the variable is bell-shaped similar to a normal or X^2 distribution which can be advantageous during the training of the model, so it was decided to treat the variable as numerical. In this way, instead of classify, the models were used for regression.

So, the models were chosen among those suitable to perform regression analysis and were K-nearest neighbors

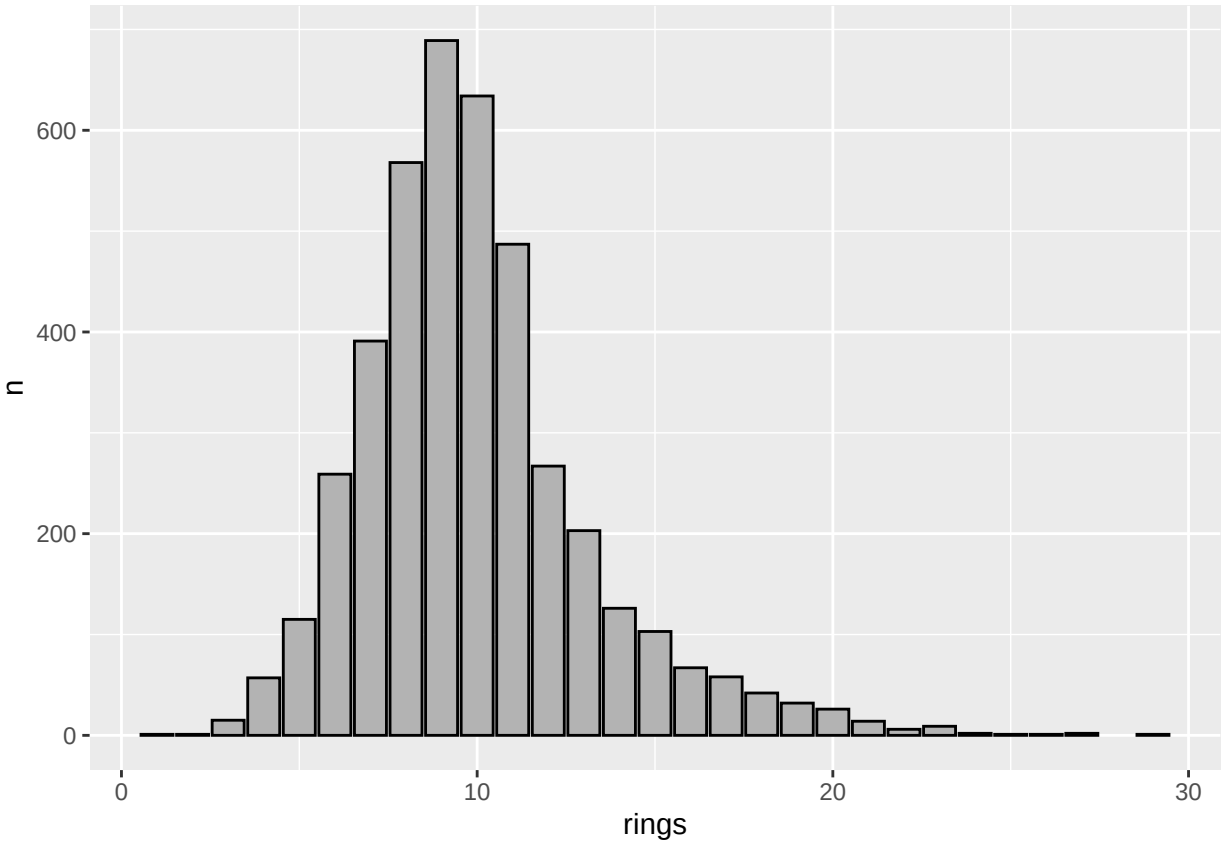

Figure 2: Frequency of different values of outcome variable

(KNN), Generalized Linear Model regression (GLM), Classification and regression trees (CART), Random forest (RF), smoothing through local weighted regression (loess) and linear support vector machine (SVML). A model was trained for each of those algorithms and a its performance was evaluated by using two metrics: the Root Mean Square Error (RMSE) and the coefficient of determination, R^2. The RMSE quantifies how deviates predicted values from the real values and it is calculated by the following formula.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}$$

The coefficient of determination provides information about the goodness of fit of a model and is useful for regression purposes. Its calculation is given by the following formula:

$$R^2 = 1 - \frac{SSR}{SST} = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

where SSR is the sum of squares of regression and SST is the total sum of squares, y_i is the actual value of a i-th instance while $\hat{y}_i$ is the predicted value and $\bar{y}$ is the mean value. The closer to one more accurate is the prediction. For the models KNN, RF and SVML, a control training was applied consisting of cross-validating training ten times by using a random splitting of 10% of the data. For the remaining models only a set of tuning parameters were evaluated to choose the one that minimizes RMSE. For the training of models, the caret package implemented in R was used.

Once all the machine learning algorithms-based trained models were obtained, we attempted to improve predictions by ensembling model-specific predictions. As a first approach, model predictions were sequentially averaged as they were trained and obtaining their respective predictions. The average predictions were submitted to RMSE validation and checked its preformance. On the second approach, the best models were chosen based on the RMSE and their performance when they were averaged in the first ensemble. Once the best algorithm models were chosen, their predictions were averaged to get a single prediction and it was checked as before.

Results

Performance of machine learning algorithm-based models

K-Nearest Neighbors For this model, the parameter neighbors or K was tuned by searching the best value from 1 to 40 to achieve the lowest RMSE. the best value was 23, as can be seen in figure 3.

The RMSE for this model was 2.2629. The general performance and the tuning parameter for this and other models can be seen in the table 2.

Table 2: Performance of the machine learning algorithm-based models used to predict the number of rings. The name, metrics and value of tuning parameters are displayed.

Model	RMSE	R_Squared	Tuned_parameter	Best_tune
KNN	2.262933	0.5157590	k	23
GLM	2.280028	0.5084155	None	none
CART	2.505987	0.4061519	Cp	0.01
Random Forest	2.252992	0.5200043	mtry	4
LOESS	2.226885	0.5310642	span	0.433333333333333
SVM linear	2.257218	0.5182022	C	1.6

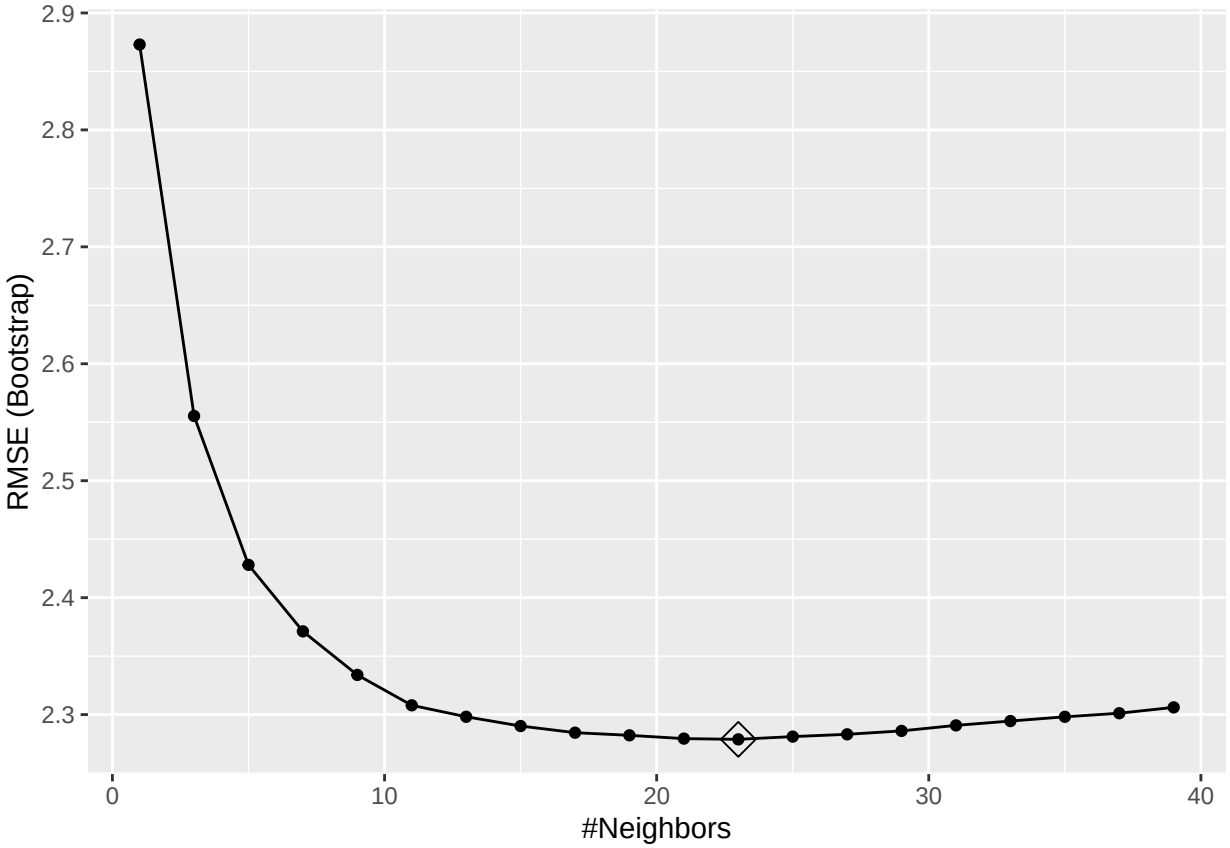

Figure 3: Performance of different values of the tuning parameter, K, for the K-nearest neighbors model

Generalized linear model This model has no a tuning parameter and with a RMSE of 2.28, it had less performance than the one seen with KNN model. However,not for this the model was discarded until its performance can be seen jointly with another model's performances and ensembles.

Classification and Regression trees The best value of the tuning parameter Cp was 0.01. This is the complexity parameter and can be understood as the minimum improvement in the model needed at each node of the resulting tree. In this case, its performance was the worst of the tested models with a RMSE of 2.5059. The output of different Cp values can be seen in figure 4.

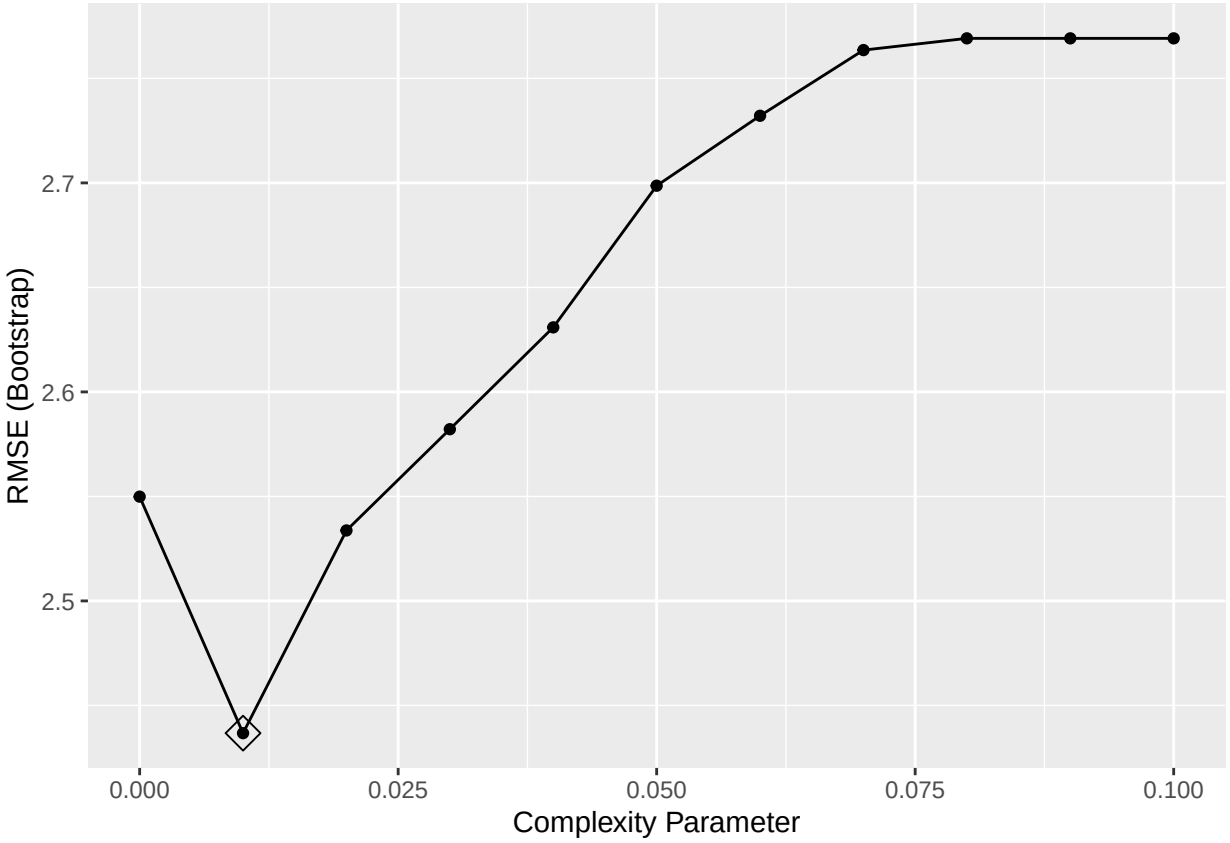

Figure 4: Performance of different values of the tuning parameter, Cp, for the Classification and regression trees (CART) model

Random Forest RMSE produced by random forest model was 2.2529, which was the second best model out of those generated by algorithms. The best RMSE was displayed by a value of 4 of the tuning parameter mtry, which is basically the number of variables randomly sampled as candidates at each split. The different mtry values can be seen in figure 5.

Generalized Additive Model using LOESS This model yields the best performs according to RMSE of 2.2268. Since it is a smoothing algorithm that can take advantage of the distributions of the variables (including the outcome variable), it is not surprising that this model performs the best. Though this model in caret package uses two tuning parameters, the degree parameter was keep it at 1, which got better estimates. On another hand, the tested tuning parameter span ranged from 0.1 to 0.7 where the best model was obtained by using a span value of 0.4333. Tuning process summary can be seen in figure 6.

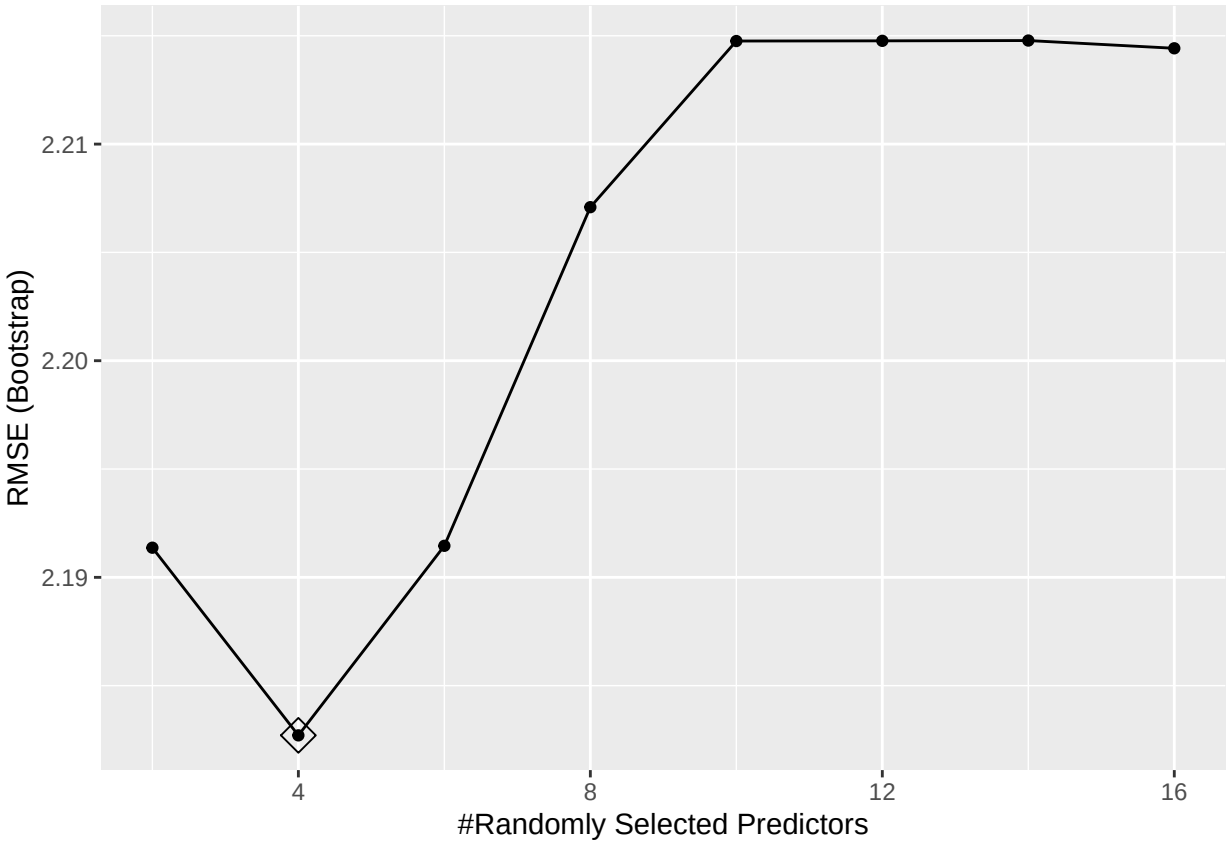

Figure 5: Performance of different values of the tuning parameter, mtry, for the random forest model

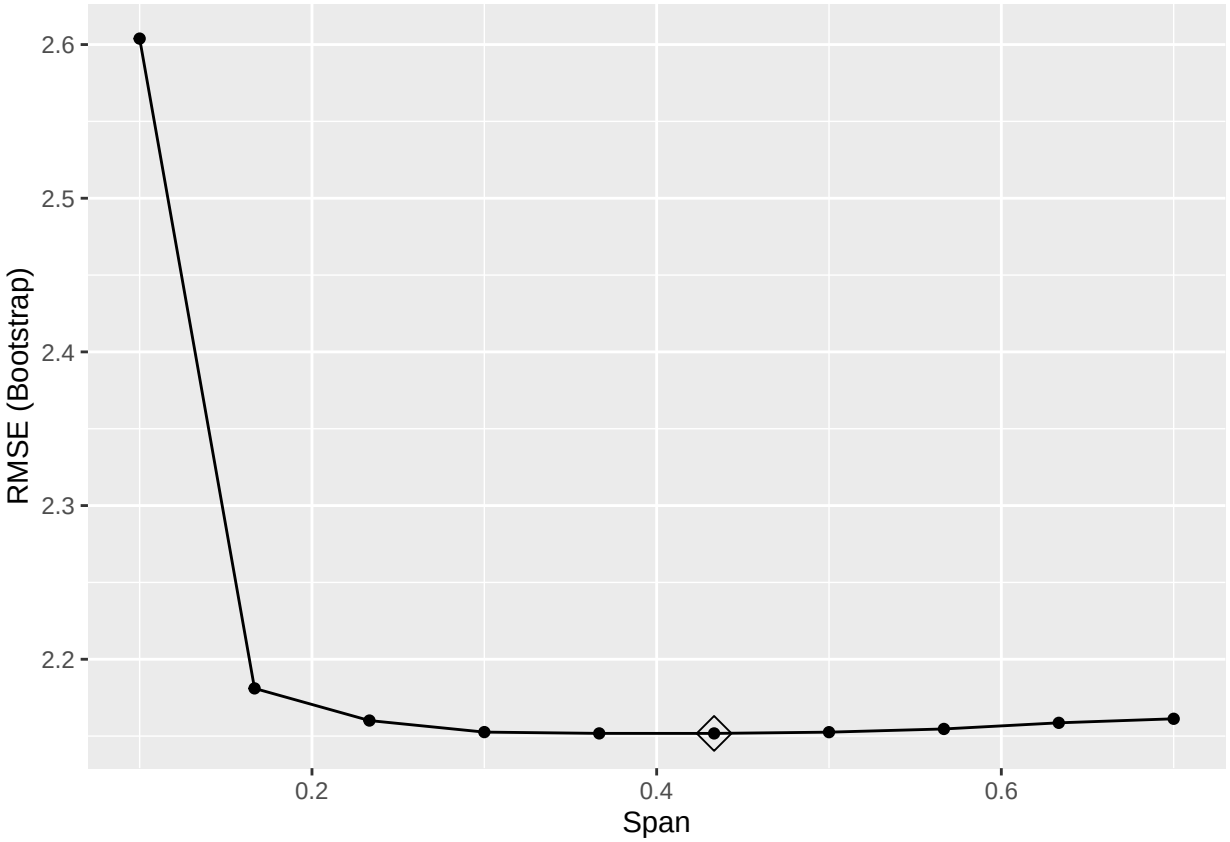

Figure 6: Performance for the generalized additive model using loess of different values of the tuning parameter, span, while kept parameter degree setted at 1

Support vector machine with linear kernel The best model runned with this algorithm yielded a RMSE of 2.2572, as can be seen in table 2. The results of the tuning parameter testing can be summarize in figure 7. In this process, different values of parameter C were tested. This parameter C, which is short for "Cost", impose a penalty to the model for making an error, so the higher the value of C, the less likely it is that the SVM algorithm will misclassify a point. In this way, the best RMSE value was achieved with a C parameter value of 1.6.

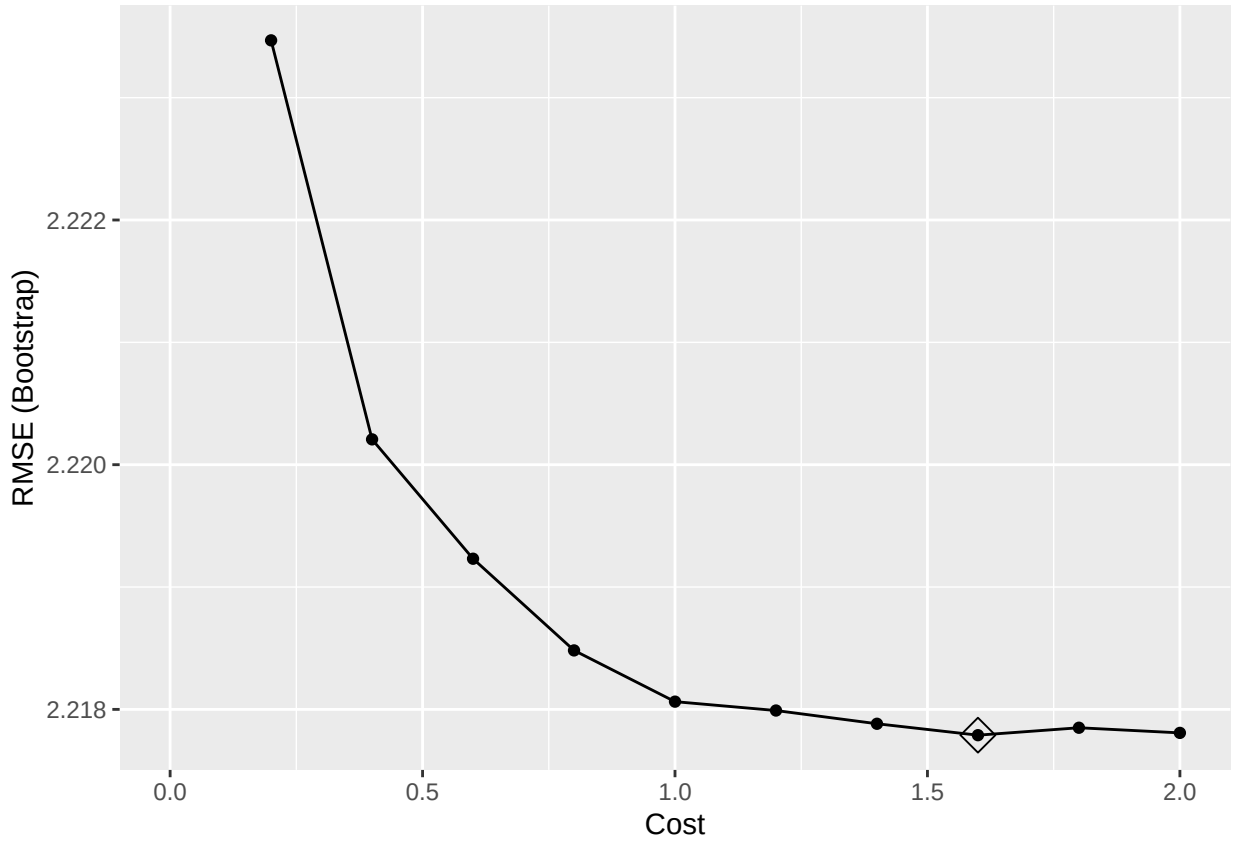

Figure 7: Performance of different values of the tuning parameter, C, for the support vector machine with linear kernel algorithm

Ensembles

Results of differente ensembles improve the results of the predictions. For the first approach used to build ensembles, the best RMSE was reached when KNN and GLM were used and the best second ensemble was the one where all models predictions were included, as can be seen in table 3.

Table 3: Result of consecutive ensembles built by sequentially adding predictions of algorithm-based models

Ensemble	RMSE	R_Squared
KNN - GLM	2.185430	0.5483607
KNN-GLM-CART	2.223410	0.5325265
KNN-GLM-CART-RF	2.208766	0.5386638
KNN-GLM-CART-RF-loess	2.191277	0.5459406

Ensemble	RMSE	R_Squared
KNN-GLM-CART-RF-loess-SVML	2.185709	0.5482452

Though the predictions improved, this process can shown that the addition of predictions from certain models worsens the RMSE. For example, addition of prediction of regression trees, lead the RMSE from 2.18 to 2.22. For this reason, taking this results jointly with individual algorithm-based results of RMSE, four models were selected for an additional ensemble. The models were KNN, RF, LOESS and SVML. An ensemble built by using these four models, named "final ensemble", performs the best by reaching a RMSE of 2.16 (table 4).

Table 4: Result of ensembling the machine learning algorithm models with better performance

Ensemble	RMSE	R_Squared
KNN-RF-loess-SVML	2.167594	0.5557026

Validation test

Once the best ensemble was chosen in order to reach the best prediction, it was used on the validation test that comprises 20% of the whole dataset (837 instances) and that it was not have been used during the previous process. In this way, the final ensemble performed extraordinarily well by reaching a RMSE of 2.029.

Table 5: Result of applying the final ensemble model on the validation test

Model	RMSE	R_Squared
Final Ensemble	2.029241	0.5568214

The relationship between the actual and predicted values of validation dataset can be explored in figure 8.

Conclusions

By taking the individual machine learning algorithm models, the generalize additive model using loess yielded the best performance, which can be explained by its ability to take advantage of the distributions of the variables.

In spite of the well performance of some algorithms, the best performance is achieved by using ensembles of the predictions given by individual algorithm-based predictions.

Not all individual algorithm-based predictions should be used in ensembles to reach better and accurate results. Though some of them takes its time to run, it is recommendable to ignore those that performs worst.

The rings of abalones, as an approximation of their age can be accurately estimated by using physical traits other than the direct counting of their rings, which is expensive, complicated and time-consuming.

Having in mind that the range of rings spanned by the entire dataset of Abalones goes from 1 to 29, a RMSE of 2.02 is a very good approximation that supports the replacement of old-fashionist and expensive methodologies for fast, easy and economical methodology to estimate the number of rings of abalones and subsequently its ages.

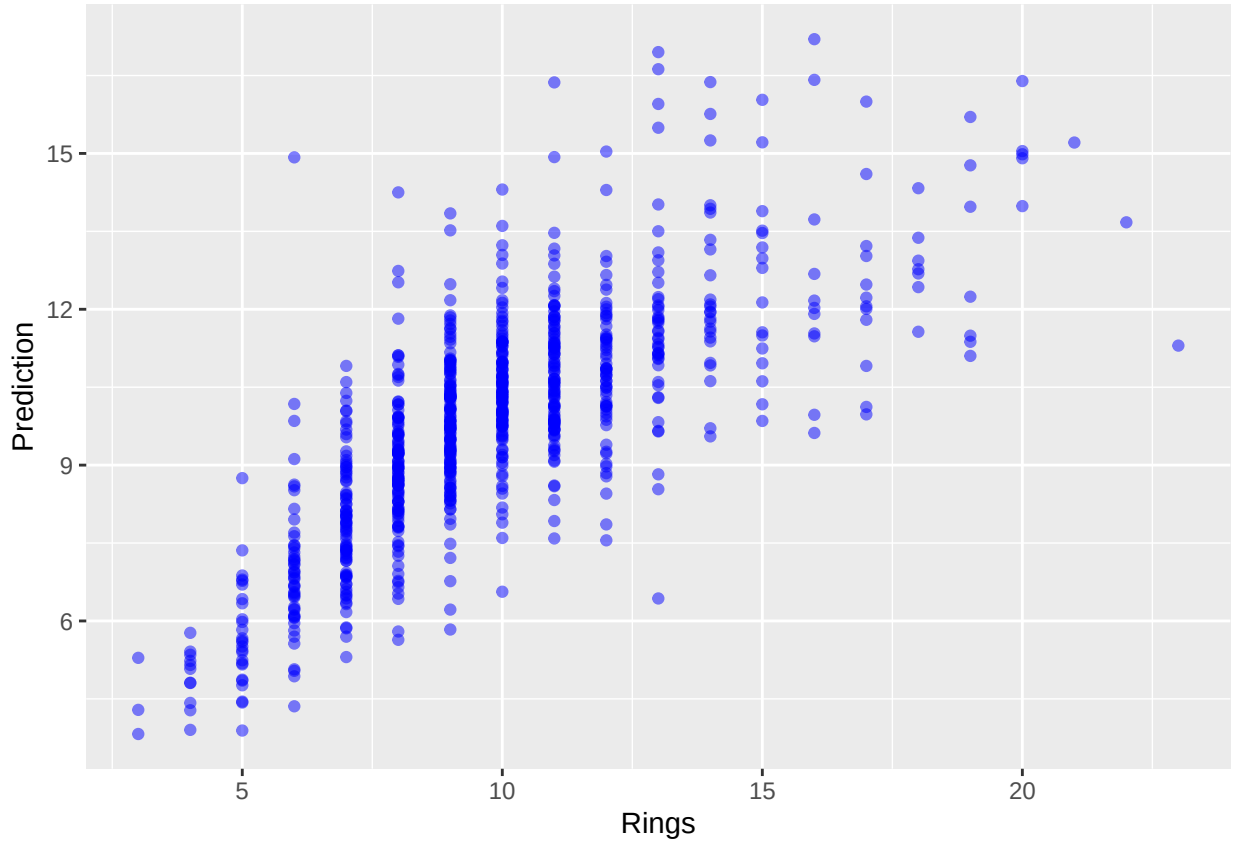

Figure 8: Relationship between actual and predicted rings according to the final ensemble by using physical traits of the instances included in the validation dataset

Acknowledgements

I would like to acknowledge to all the team of teachers and co-workers of the data Science Professional Certificate Course at Harvard University for all the learnings along this course. I enjoyed the process a lot.