Predicting House Prices in Toronto: A Machine Learning Approach

Albina Cako, BSc *York University, Certificate in Machine Learning*
Colin Green, BSc *York University, Certificate in Machine Learning*
Lucy Zhang, BSc *York University, Certificate in Machine Learning*
Sean X. Zhang, MSc *York University, Certificate in Machine Learning*

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Keywords: house prices, machine learning, caret, shiny

Introduction

The House Pricing Prediction app is created for estimate the house price for both buyer and seller based on different factors such as total Sqft, house locations, etc. The deployment was constructed using ShinyApp. An user friendly app for both buyer and seller, with simple click of factors users will get an estimated housing price. The app can be used for individual buyers who want to know the final price of the houses they are interested or for individual sellers to know what is the best listing price. The app uses regression model for prediction, which was trained by the data set of Toronto housing price. The housing price is strongly correlated with other factors, for increasing the model accuracy decreasing errors, it is important to try different factors and combinations. This project will comprehensively validate four different models: decision tree, random forest, K nearest neighbors, and gradient boosting machine. This report will go through data analysis, modeling implementation and provide an optimistic result for housing price prediction. ## Objective The objective of this project was to evaluate the application of machine learning algorithms to predict house prices in the Greater Toronto Area, and apply

Background

Purchasing a house if a big life decision for every individual and needs a considerable amount of research. Everyone has different purpose of buying houses, someone would prefer by the house at the best rate for living now, someone would buy houses for future investment. Selling the houses is also very important and needs to do research and decide what is the best leasing price. Commonly, people will ask advise from various websites, real estate agents or realtors before purchasing or leasing; However, due to the trend towards big data, house pricing prediction can be done by using machine learning strategies base on large amount of data from previous years more correctly. House Price Index (HPI) can measure the price changes of residential housing as a percentage change, In Canada the new Housing Price Index is calculated monthly by Statistics Canada. HPI is useful but because it is a rough indicator calculated from all transactions, it is

inefficient for predicting a specific house with its attributes. The purpose of this project is to create an app for both buyers and sellers can easily check the predicted list price or final price based on the attributes of the house such as locations, square foot, number of bedrooms, etc.

Methodology

Data Preprocessing

The housing dataset, originally shared on Github[1], was extracted from Zoocasa.com in the summer of 2019. The dataset contained all completed property sales in the city of Toronto within a 1-year span. We performed several data exploration and cleaning steps to prepare this data for modeling.

Missingness

We assessed the dataset for missing values. Many parametric machine learning models do not accept missing data, and the accuracy of even non-parametric models are often negatively impacted by missingness. Thus, missing values ought to be either imputed or removed before data modeling. We then determined whether missing data was Missing Completely at Random (MCAR), Missing at Random (MAR), or Missing Not at Random (MNAR). Should the data be MCAR, then it is acceptable to simply remove each observation that is missing, as doing so would not introduce bias to the remaining observations. However, if there was a correlation between missingness and other data features, then imputation must be performed. Missingness correlation was assessed using the missing_compare() function from the finalfit library, which applies the Kruskal Wallis correlation test for numerical variables and chi-squared correlation test for categorical variables to determine correlation. Using the MICE package in R, we then applied the following imputation methods: 1) simple, which imputes a value from a simple random sample from the rest of the data; 2) mean, which imputes the average of all observations; 3) random forest, which applies a random forest algorithm; and 4) CART, which imputes by classification and regression trees. The distribution of the imputed data were evaluated with a density plot and an imputed dataset was chosen based on best fit.

Assessing Parametric Fit

Outliers were visualized with the boxplot() function, with outliers falling outside Q1 - 1.5 X Inter-Quartile Range and Q3 + 1.5 X Inter-Quartile Range. Normality of the distribution of variables were visualized with density plots. A correlogram with Pearson's R determined collinearity. Linear relationship between outcome variable and predictors was tested via scatterplots.

Data Curation

The following variables were removed as they did not have any data utility or had free-text, which would have necessitated additional cleaning: title, description, mls, type, full_link, full_address. A numeric 'bedrooms' column was created by combining bedrooms_ag and bedrooms_bg. We also removed district_code and city_district; both were a categorical variables with number of factors = 140; keeping it would significantly increase model training time. Longitude and latitude were also not considered to accommodate model deployment, as including these variables in training sets would have required geocoding and assignment of latitude and longitude inputs

to districts; tasks which were outside our scope for this application. Mean_district_income was left as an approximation of the effect of districts on the house price. After consultation with a real-estate expert, we decreased the number of property types by generalizing from 10 different types to: Townhouse, Condo, Detached, Semi-Detached, and Plex

Thus, the predictors chosen were:

Table 1: Predictor variables

Label	Description
sqft	numeric
beds	numeric
bathrooms	numeric
parking	numeric
mean_district_income	numeric
type	categorical

The target variable chosen was final_price. Rather than training two models to predict on both list and final price, the predicted list price was instead approximated by a linear equation between list and final price in the training data.

Modeling

The data contained a mix of categorical and numerical variables and did not satisfy many requirements of parametric models, such as variable independence, and normally distributed data. Thus, several parametric models were used. We trained four different non-parametric models using k-fold cross validation. The models were then tuned using various grid searches to improve the accuracy. The final model was chosen based on three metrics: Root Mean-Squared Error (RMSE), Pearson correlation (R^2), and Mean Average Error (MAE).

Deployment

The application was created using R shiny. The user interface (UI) contains a map of Toronto generated by the leaflet library for geographic navigation, and allows the user to select various inputs to predict property price. While the user would choose a district of interest from the front end, the back end links the district chosen with income and uses mean_district_income as the model input instead. We chose INSERT_MODEL HERE, since it was the most accurate model as the back-end for our application.

Results

The original housing dataset contained 21 variables and 15234 observations. Table 1 defines each variable of the dataset.

Table 2: Data Dictionary

Label	Description
title	Title of the listing
final_price	Final price of the property
list_price	Listing price of the property
bedrooms	Number of bedrooms
bathrooms	Number of bathrooms
sqft	Area of property in square feet
parking	Number of parking spaces
description	Verbatim text description of the property
mls	MLS ID
type	Property type
full_link	URL to listing
full_address	Full address of the property
lat	Latitude
long	Longitude
city_district	Toronto district to which property belonged to
mean_district_income	Average household income of district
district_code	Numerical code of the district
final_price_transformed	Box-Cox transformation of final price
final_price_log	Log transformation of final price
bedrooms_ag	Number of bedrooms above ground
bedrooms_bg	Number of bedrooms below ground

Data Exploration

Discussion

Acknowledgements

The authors would like to thank Hashmat Rohian, adjunct faculty at York University for supervision of the project. We also thank Slava Spirin for the original extraction of the Toronto Housing dataset [1]. Finally, we thank Steve V. Miller for creation of the manuscript template in R Markdown [2].

References

1. Spirin, S. (2020). Slavaspirin/toronto-housing-price-prediction Available at: https://github.com/slavaspirin/Toronto-housing-price-prediction [Accessed October 26, 2020].

2. Miller, S.V. An r markdown template for academic manuscripts. Steven v. Miller. Available at: http://svmiller.com/blog/2016/02/svm-r-markdown-manuscript/ [Accessed October 26, 2020].