Madelyn Goodman | madelyn.b.goodman@gmail.com
Brainstation Winter 2022 Capstone Project | April 4, 2022

CryptoPunk Price Predictions Using Machine Learning

NFTs, CryptoPunks, and Why Machine Learning Can and Should be Utilized in the Space

NFTs, or non-fungible tokens, experienced a 20,000% increase in trading volume from 2020-2021 with an overall market value of $17 billion making them one of the fastest growing digital assets. So far, machine learning and AI technology has mainly been utilized to generate these pieces of digital art with very few projects harnessing these methods to predict market trends, making these models a largely untapped resource for investors and traders.

In 2017, the NFT project, CryptoPunks was launched by Larva Labs as a set of 10,000 free tokens associated with unique pictures computer generated of figures. Since, these pieces of digital art have accumulated a trading volume of $1.7 billion. The CryptoPunk project is unique in that there was only one drop of these NFTs with a fixed set of characteristics of varying rarities making predicting their sale prices an interesting exercise to apply machine learning to as the scarcity of the asset is fixed. A unique feature of the application of blockchain technology is that all the data on transactions is stored on a publicly available immutable digital ledger, meaning that all the data on every CryptoPunk transaction is accessible to the public and can be used to inform future trading.

I wanted to look at if machine learning techniques could be utilized to predict the sale prices of CryptoPunks in order to formulate smart trading strategies. Using machine learning to predict stock prices is a common problem space for data scientists, setting a precedent for using such techniques to make predictions on the NFT market.

My Data

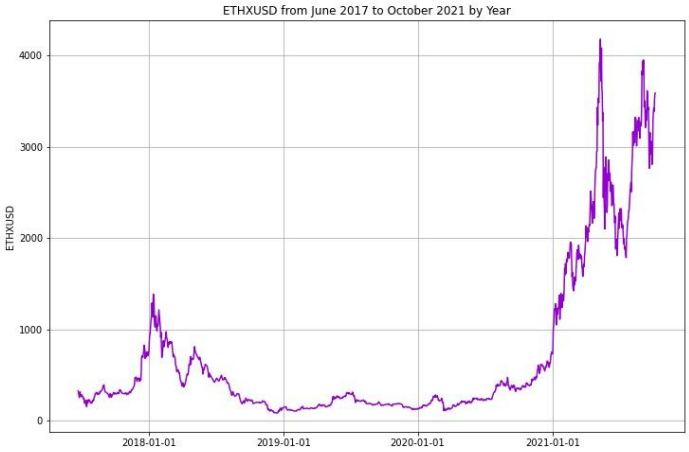
My primary source of data for this project was a dataset from Kaggle of CryptoPunk transactions compiled from Larva Labs and Opensea, one of the largest NFT trading platforms, from 2017-2021. This dataset included both numeric data on CryptoPunk trades, as well as metadata on the punks themselves including type (male, female, zombie, ape, and alien) and accessories/traits (beanies, tiaras, top hat, welding goggles, etc.). As I was interested in data related to sales, I only kept observations on bids and sales from this dataset. Much of the missing and duplicate data were associated with other transaction types, so I did not have to do much data cleaning. I analyzed the data for price outliers by looking at trends in price over time and box plots identifying those that were outside of the norm even for the most valuable punks.

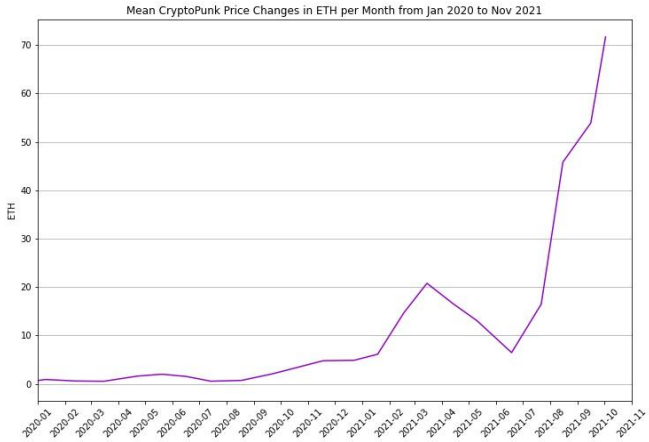
My second source of data was from Tradingview on daily market action of Ether (ETH), the cryptocurrency CryptoPunks are traded with. Only information on the date and closing price were kept from this dataset. Both sets of data were structured data stored in a tabular format and were joined using date as a key.

From this data, I created a variable of days-since-claim to account for the time a punk had been on the market at sale. To investigate how bidding action affected price I made variables of the number of bids per sale and mean bid. I also engineered variables for number of previous sales, previous sale prices, and characteristics of buyers and sellers. Finally, I vectorized the type and attributes of each punk and used the proportion of the occurrence of each attribute and type to calculate a rarity index.

Exploratory Data Analysis

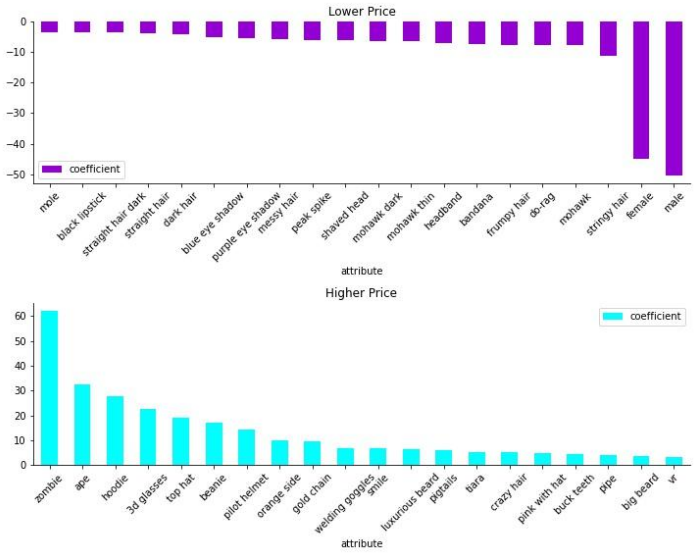
In a bull market, traders look for multipliers of a base asset. When looking at the change in price of Cryptopunks from 2020-2021, it appeared to mirror the pattern of the price movement of ETH during this time.





This was an important insight to have when evaluating my models as it emphasized the fact that the majority of CryptoPunk trading has never seen a sustained decline in the value of ETH, meaning that whatever model I trained on this data would likely not be able to be applied in a bear market.

One of my main hypotheses was that attributes of a CryptoPunk significantly impact its price. I tested this hypothesis with a crude OLS regression finding that 26 of the 87 total attributes were significantly related to price. These attributes and types showed directionality in their relation to price in an additional ridge regression analysis. Notably, being a zombie had the greatest effect on higher price while male and female punks had the greatest association with lower price. A smart trading strategy could be to look for punks with traits such as “welding goggles”, “smile”, and “luxurious beard” as they positively impact price, while not to such a great degree that they make the punk unattainable to most buyers.



I was also interested in how punk rarity was related to changes in price. Plotting the mean price of a punk against its rarity index revealed 5 distinct rarity groups prompting me to create an ordinal variable that I used to visualize rarity on a plot of change in price over time.

One surprising dynamic seen in this plot was that “less rare” and “not rare” punks had increased in price more towards the end of September 2021 compared to punks that were “somewhat rare”. This could have been due to the rarer punks having been set at a higher price prior to the NFT boom, therefore, when more buyers and sellers came into the market and more competition was introduced, those that were priced higher couldn’t be resold at a comparatively greater margin. This could also be an explanation for the apparent depreciation in some of the rarer punks during this period.

Model Building, Evaluation, and Selection

Madelyn Goodman | madelyn.b.goodman@gmail.com

Before modeling I performed a train/validation/test split and scaled the data. Several of the models I explored, ridge and support vector regression as well as neural networks, responded best to scaled data. I chose a standard scaler as opposed to a minmax scaler so that future scaled data wouldn't be restricted between the minimum and maximum of the training data.

I started the model building process with a GridSearch of decision tree, ridge, and support vector regressors. The decision tree regressor performed the best of the three with a max depth of 9 and an R^2 of 0.79, meaning the model fit 79% of my validation data. This led me to test a random forest regressor, as it is a bagging method on decision trees, aggregating trees fit to random subsets of data (in my case 50), creating a better fitting model. I used the hyperparameter of a max depth of 9 found in the GridSearch and fit it to unscaled data, as random forests do not require scaled data. This model performed extremely well with an R^2 of 0.84.

As neural networks are widely regarded as the most robust predictive modeling technique I also built several networks to compare the performance to the random forest regressor. I designed the architecture of these models by choosing the number of nodes per layer based on there being 101 total features, 92 traits and types, 87 traits, 5 types, and 8 numeric features. Testing different combinations and ordering of these layers and dropout layers, I compared the accuracies of the 5 best performing models. Keras as a package introduces a certain amount of randomness when building neural networks even when setting random states, therefore, I ran each model 15 times, taking the average of the R^2 scores per model. The best network had an average R^2 of 0.68 on the validation data.

	eth	predicted_eth
0	25.25	25.542993
1	25.75	27.817275
2	31.90	34.559022
3	79.00	71.327791

The random forest model out performed the neural network with an R^2 of 0.84 on the test data. To demonstrate how closely this model was able to predict punk prices, I ran the model on data for punk #8565. The model was able to get very close to predicting the price of each sale giving me further confidence in the applicability of this model.

Conclusions

In my analyses I was able to prove my hypotheses on the impact of punk attributes and rarity on price. Further, I fit a random forest model that predicted CryptoPunk prices with 84% accuracy. This model can be used with relative confidence to form smart trading strategies for CryptoPunk traders.

As previously mentioned, using data only from a bull market can only predict prices in a bull market. In order to make this model applicable in the long term I would like to connect it to the Etherscan API to make regular calls of data to be able to build a more dynamic model that can adapt to shifts in an ever changing market. This would also allow the model to take varying transaction fees into account, a limitation of my data. Further, I would like to explore doing more time series analyses to make a model that can extrapolate price further into the future.