Machine Learning for NFT Price Predictions

Mingxuan He

Booth School of Business, University of Chicago
mingxuanh@uchicago.edu

November 30, 2023

Primer on NFTs

- NFTs are unique digital assets (e.g. photos) stored on a blockchain.
- Mostly transacted using cryptocurrencies, predominantly Ethereum.
- Most NFTs are sold on "marketplaces", grouped by "collections"

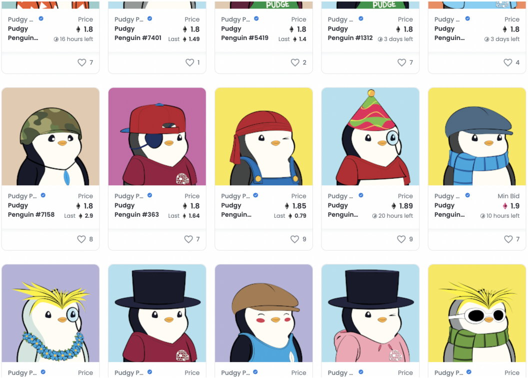

Figure 1: The Pudgy Penguins collection on OpenSea marketplace.

Motivation

Various applications of NFT pricing models: Trading, decentralized borrowing/lending, tokenized real-world assets.

Literature:

- NFT prices co-moves with market trends (Jain et al., 2022; Kapoor et al., 2022; Kong & Lin, 2021; Nadini et al., 2021)
- High rarity drives up NFT prices (Kong & Lin, 2021; Mekacher et al., 2022)

Project goal:

- Build a ML model for a specific collection to predict the price of NFTs in real-time.
- Incorporate data on market prices and individual rarity.

Data Sources

- **Dune Analytics**[1]: Public database of NFT trades on Ethereum.
- **OpenSea API**[2]: Data from largest NFT marketplace on traits and rarity for each NFT in their collections.

[1] https://dune.com
[2] https://docs.opensea.io/reference/api-overview

Data Features

Market features (by collection, daily)

- Trading volume
- 5% price
- Highest price
- Lowest price

Traits (by NFT)

- Background color
- Eyes
- ... (7 total)

Last sale (by NFT)

- Price of last sale
- Time since last sale

Rarity Rank (by NFT)

- Rarity rank within the collection (computed using OpenRarity Standard)

Price history (Bored Apes)

Figure 2: Bored Apes: Historical trades.

PCA analysis

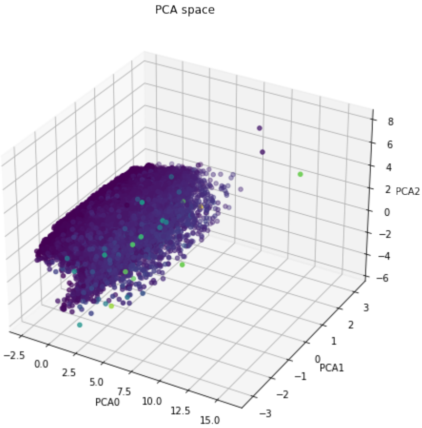

Figure 3: Bored Apes: Top 3 principal components.

Baseline Models

Framework: Python `sklearn`

- Linear models: OLS, Lasso, Ridge
- Tree-based models: Random Forest

Deep Learning Model

Framework: Python `tensorflow.keras`

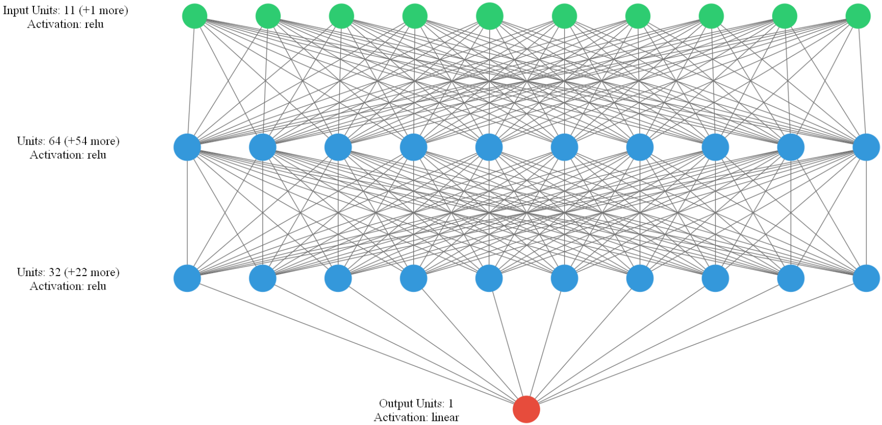

Figure 4: Neural network architecture

NN Training

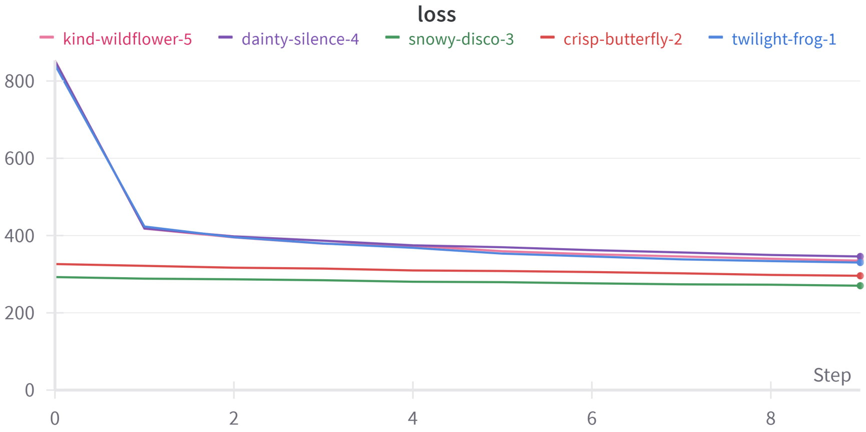

Figure 5: NN: Training loss by epoch

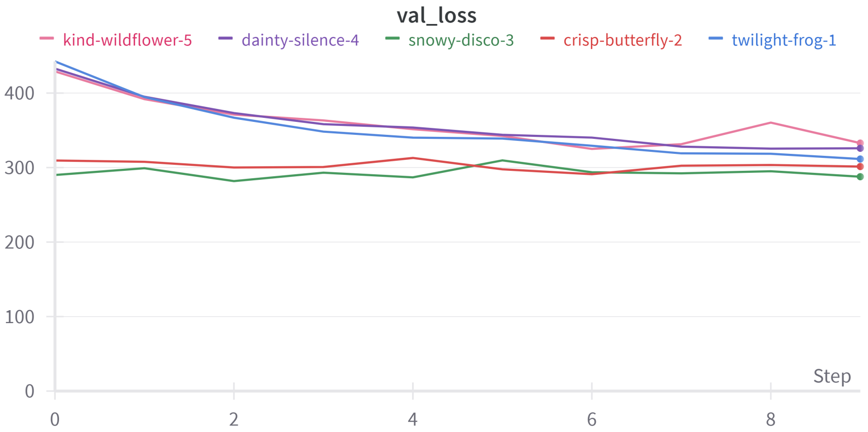

Figure 6: NN: Validation loss by epoch

Results

Current performance: tree-based models > neural net > linear models

Model	MSE loss
OLS	455.8
Lasso	467.7
Ridge	455.8
Random Forest	219.4
Gradient Boosting	209.1
FF Neural Net	287.8

Next Steps

- Test modeling pipeline on more collections (Punks, Penguins, Ninjas, etc.)
- Implement more neural nets (RNN/LSTM, etc.)

Github repo: https://github.com/mingxuan-he/NFT-pred