PREDICTING THE RENT OF BROOKLYN APARTMENT LISTINGS ON STREETEASY

Nicholas Dell'Aquilo

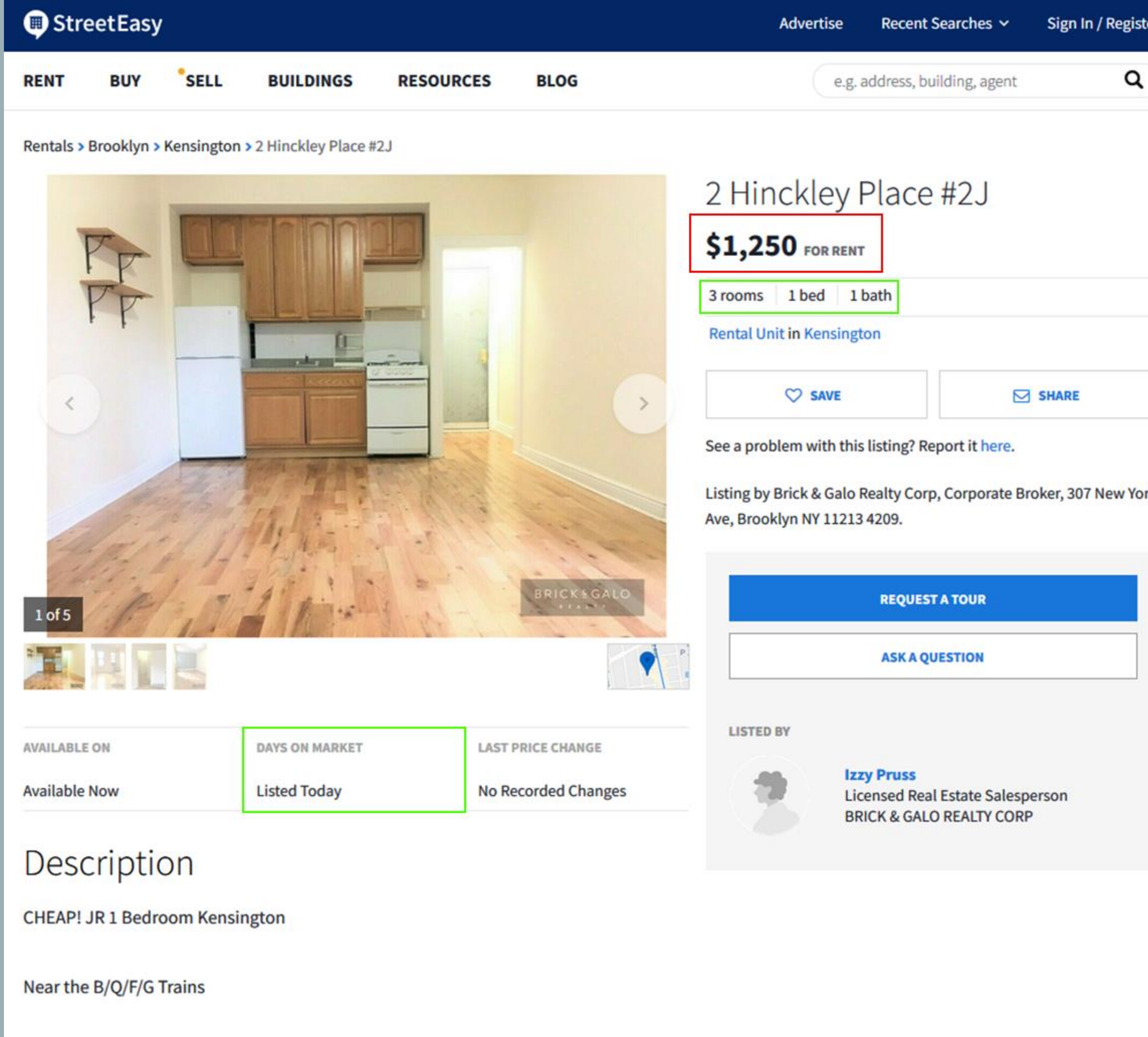
BACKGROUND

- Apartment listings on StreetEasy

- Listings include details on the features and amenities available

- Is it possible to predict the rental cost of an apartment with the information available?

- This could be used to negotiate for a better price, or know when a good deal is available



DATA

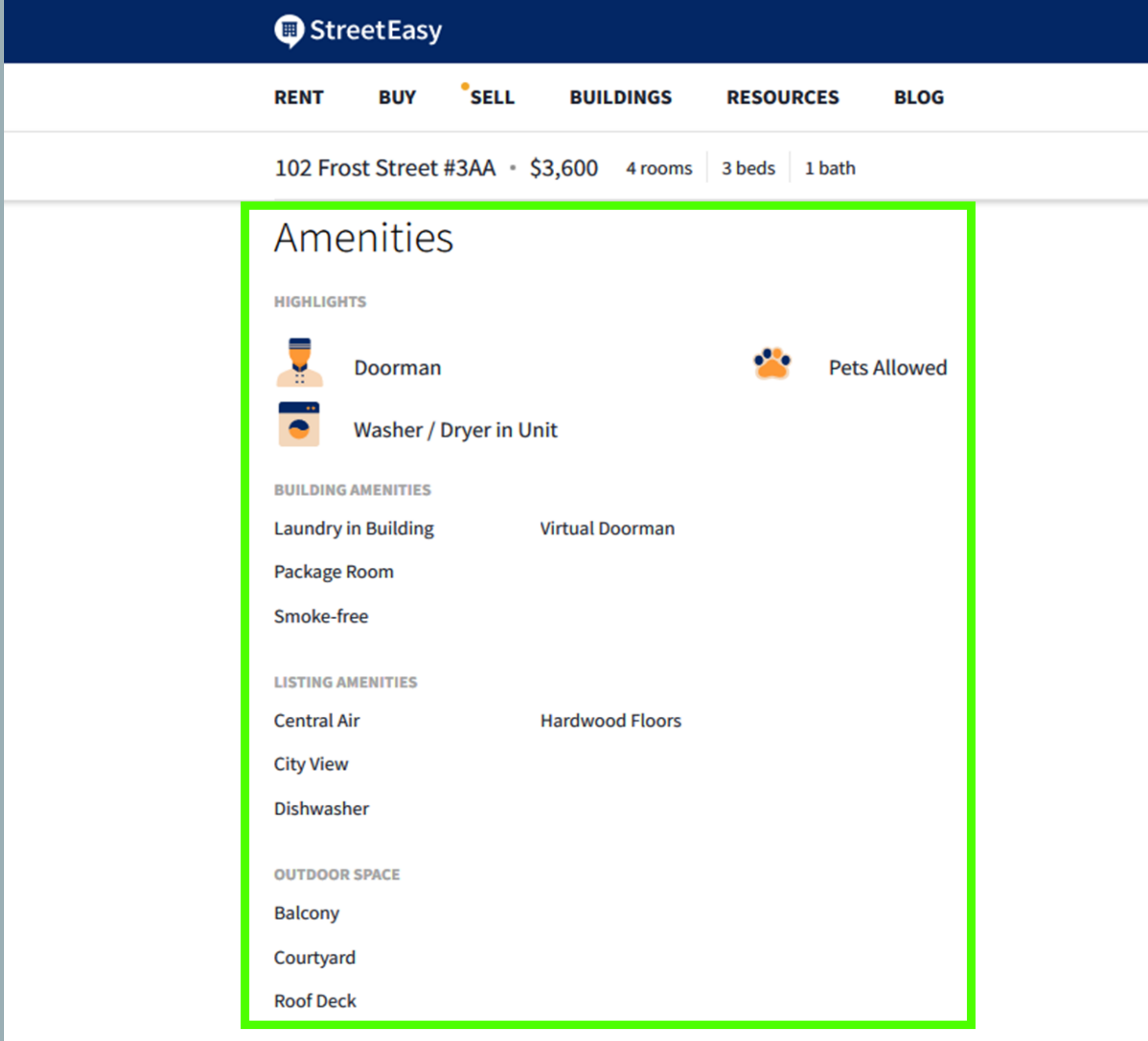
- 3,568 listings scraped from StreetEasy

- Removing null values reduced observations to 2,839

- Removing outliers (Target or feature variable >3 standard deviations more/less than mean) reduced observations to 2,758

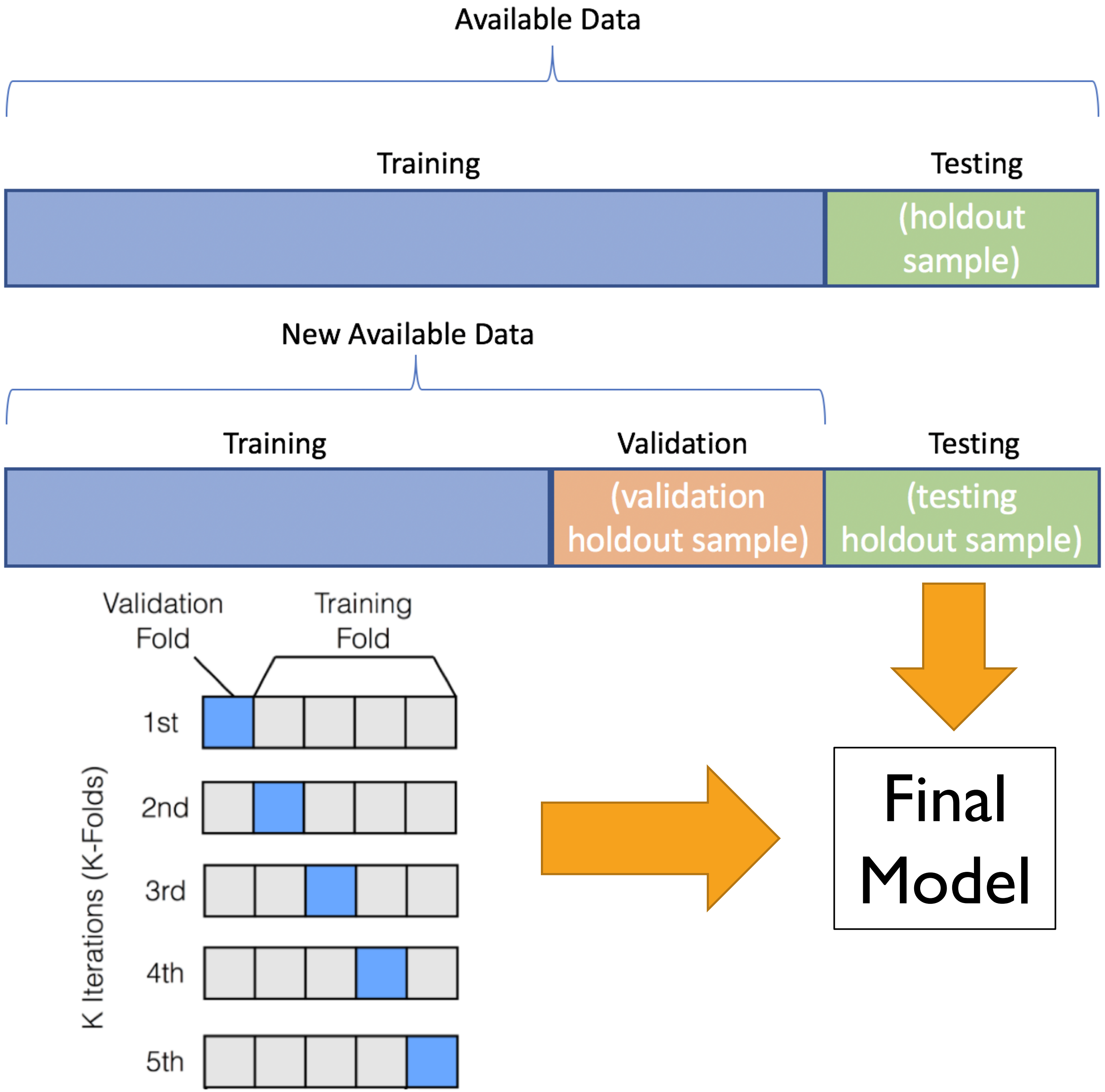
- List of amenities converted into binary dummy variables

    - 61 initially, reduced to 49 through removal of outliers



MODEL SELECTION PROCESS

- Dataset split into 60/20/20 train/validate/test partitions
- R^2 calculated in 5-fold cross validation on training partition
- Model fit on validation partition and compared with training scores
- Once all models trained, validation scores compared to determine best model
- Final model re-trained using test and validation partitions, compared against prediction of holdout test data



SIMPLE LINEAR REGRESSION (BASELINE)

- Objective: test the simplest possible features that someone might consider

- Total rooms, bedrooms, bathrooms, and whether there is a studio

- R^2 on training partitions ranged from 0.18-0.31, mean of 0.25; high variance

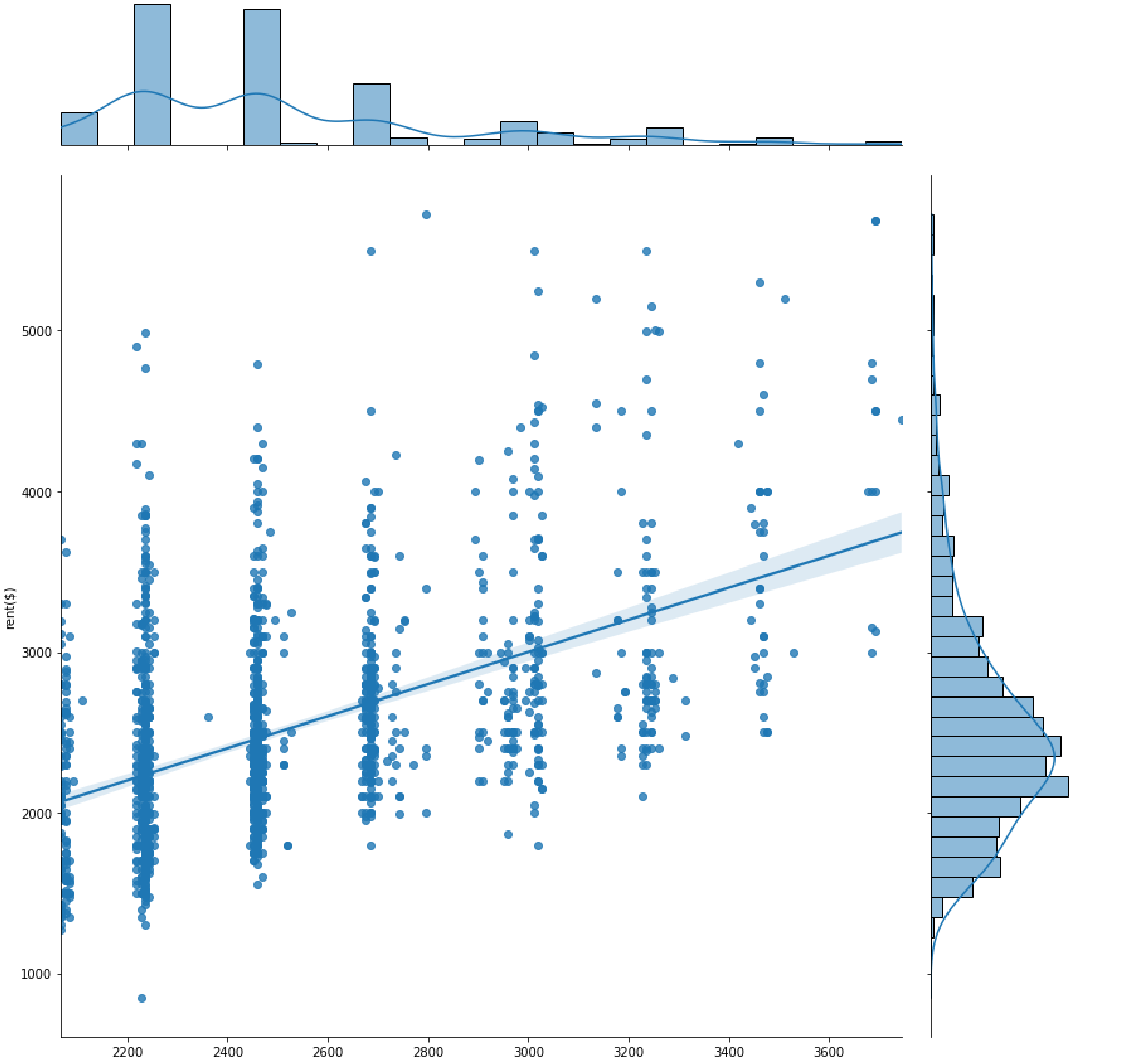
- R^2 of validation: 0.24

- The mean model isn't overfitting, but is probably underfitting

-Intercept: $1,443

-Coefficients:

  - Rooms: 8.37
  - Bedrooms: 216.24
  - Bathrooms: 551.15
  - Studio: 65.96

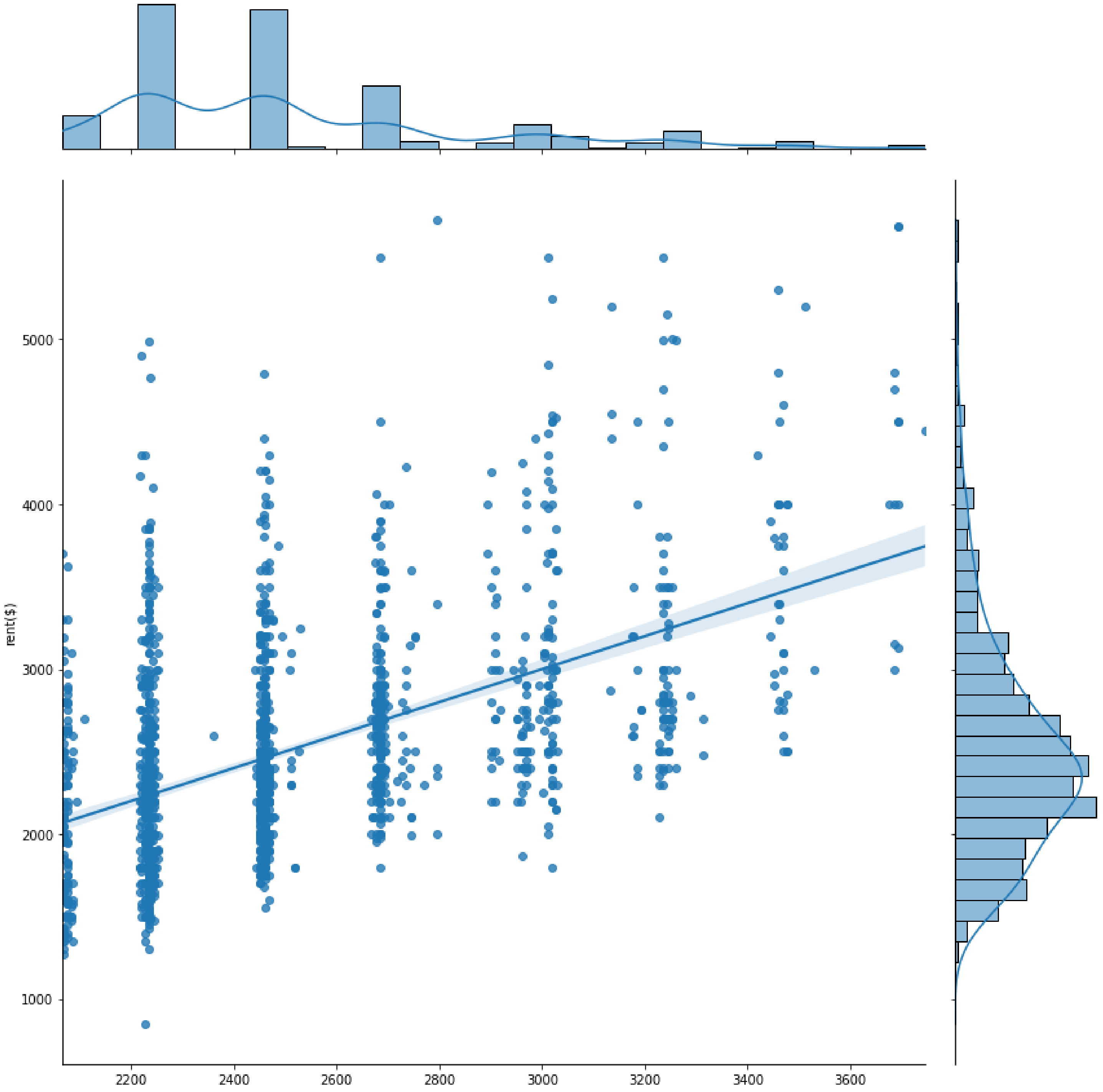


LINEAR REGRESSION WITH NEW FEATURE

- Objective: Add a new feature (number of days on market) to determine if it has an impact on the model

- Result: almost **no impact** on model

- Coefficient of new feature: -0.14



LASSO REGRESSION WITH NEW CATEGORICAL FEATURES

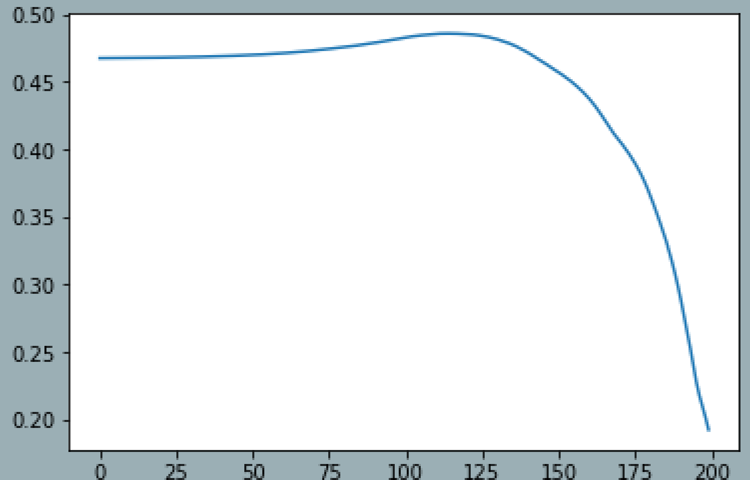
- Objective: Add many new, categorical features, handle potential multicollinearity with Lasso

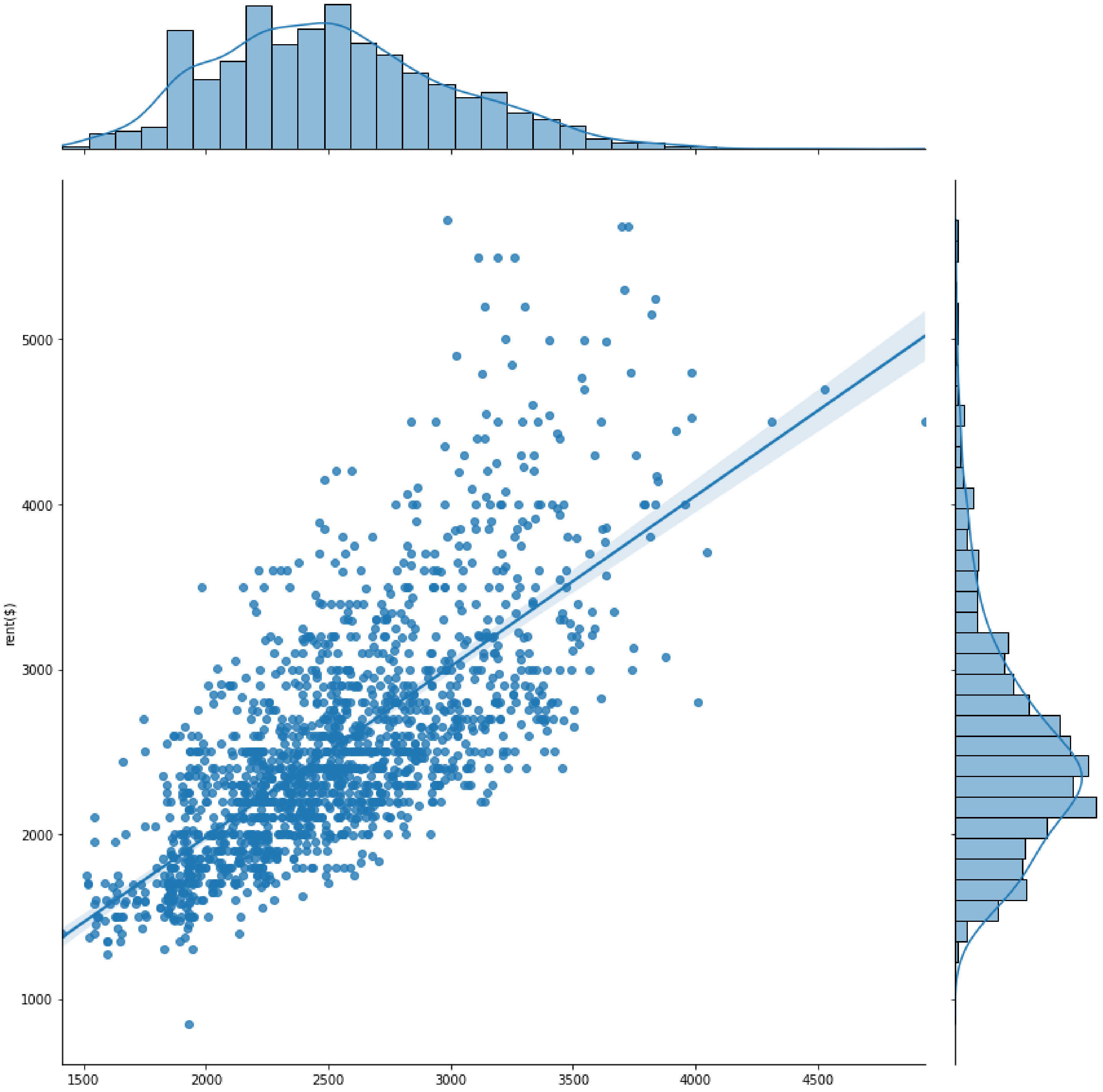
- 49 new binary categorical features, representing whether an amenity (e.g. Balcony, Air conditioning, etc.) is present

- 200 alpha values for Lasso tested in cross-validation, best one used (1.96)

- Mean R^2 = 0.49, ranges from 0.45-0.53

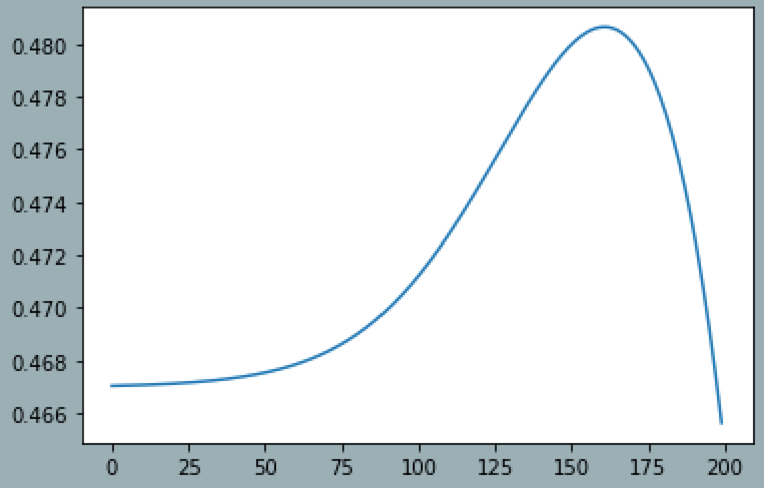
- R^2 on validation = 0.47; not bad!

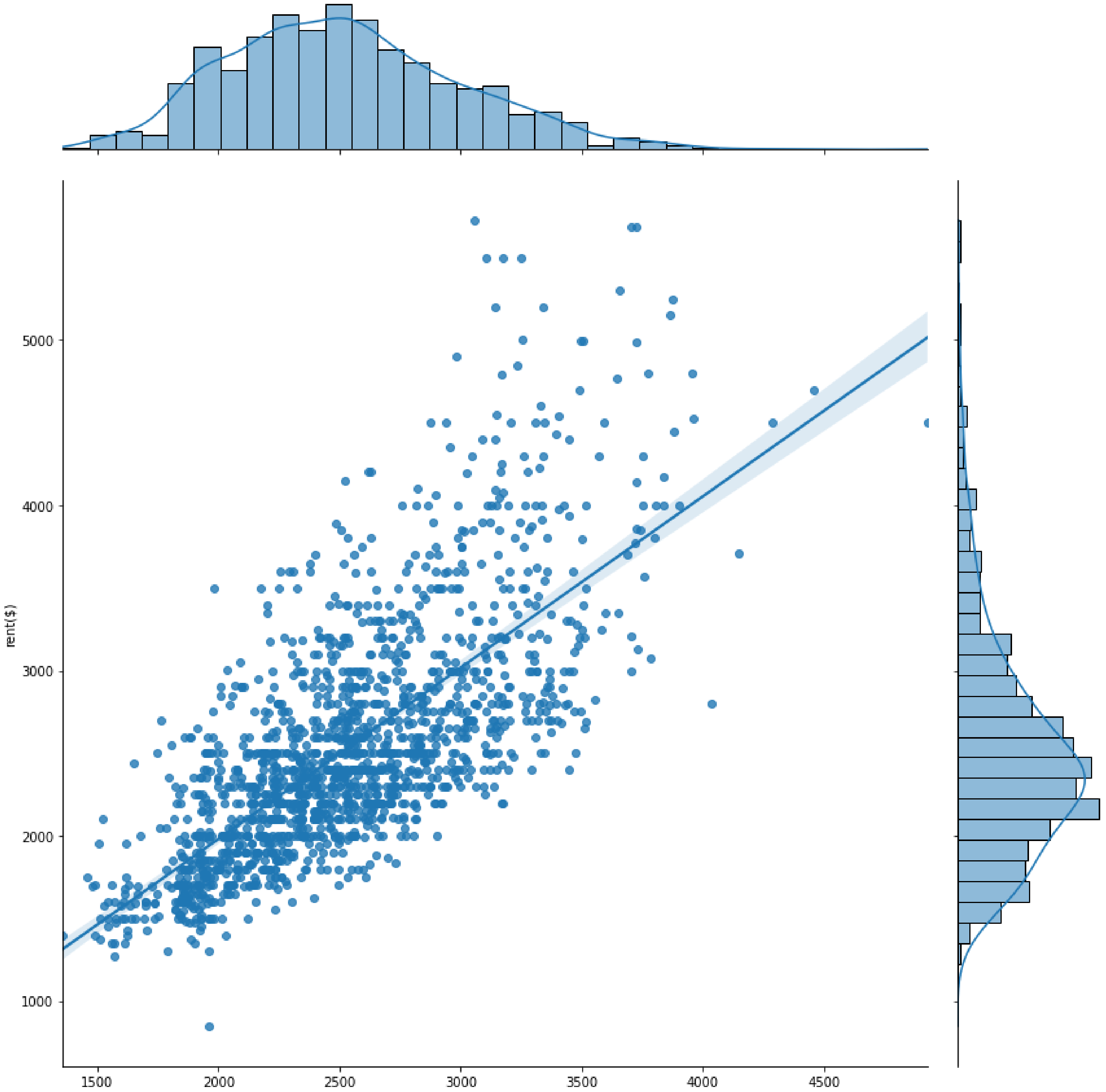




RIDGE REGRESSION

- Objective: Compare with Lasso to determine if removing features hurts the model
- Same features as Lasso model
- 200 alpha values tested in cross-validation, best one used ()
- Mean R^2 = 0.48, ranges from 0.45-0.52
- R^2 on validation = 0.46 (~0.0011 less than lasso)





COMPARING MODELS

- R^2 scores on validation partitions:
  - Linear Regression: 0.24
  - Lasso: 0.4655
  - Ridge: 0.4644
- Lasso is the best model, just barely. Retraining the model including the validation partition, R^2 is…
  - 0.48!
- Compared to the test scores, it's unlikely that this model is overfitting
- Intercept = $1,356.78

TOP COEFFICIENTS

Feature	Coefficient
Concierge	406.219398
Water View	388.498671
baths	329.212511
Washer / Dryer in Unit	316.736976
beds	270.995151
Fireplace	247.607404
Storage Available	237.065863
Cats and Dogs Allowed	216.009136
Dishwasher	179.272271
Locker/Cage	158.05087
Gym	148.249338
Roof Deck	129.101006
Package Room	121.702767
Terrace	108.557168
Garden	105.69227

THOUGHTS & FUTURE IMPROVEMENTS

- R^2 was somewhat low, but there is a demonstrably measurable correlation between the features and target variable.

- How to improve the model?

  - More features. The model isn't overfitting, and there is clearly some aspect that is affecting rent price outside of the selected features.

  - More data. Less than ½ of the total listings for rentals in Brooklyn on StreetEasy were able to be scraped.

  - Polynomial model. The model visualization (as well as common intuition about how people value homes) hints that there may be some feature interaction.

THANK YOU

Tools Used:

- **Web scraping:**
  - BeautifulSoup
  - Selenium
  - Pandas
- **Regression:**
  - NumPy
  - SciKit-Learn
- **Visualization:**
  - Matplotlib
  - Seaborn