House Price Prediction

## Motivation:

As international students, one of the most important things for us here in the USA is housing. So, we thought of analyzing the housing prices by collecting the data from the available open sources. This project helps us and our fellow students to get insights about the Housing price trends year wise in the USA cities. So, we are planning to build and deploy linear regression model to predict house prices of USA States using Google AI platform.

## Project Architecture:

The following is the workflow of the developed scikit-learn linear regression predictive model that we followed and deployed using the Google Cloud Platform.

Training the model

Coding the model

Prepare the source data

Deploying the trained model

Getting the prediction

Managing the model

# Steps of building the project Step1: Collect the data.

Here we collected the house price data from the Kaggle website. Link: <https://www.kaggle.com/shree1992/housedata>

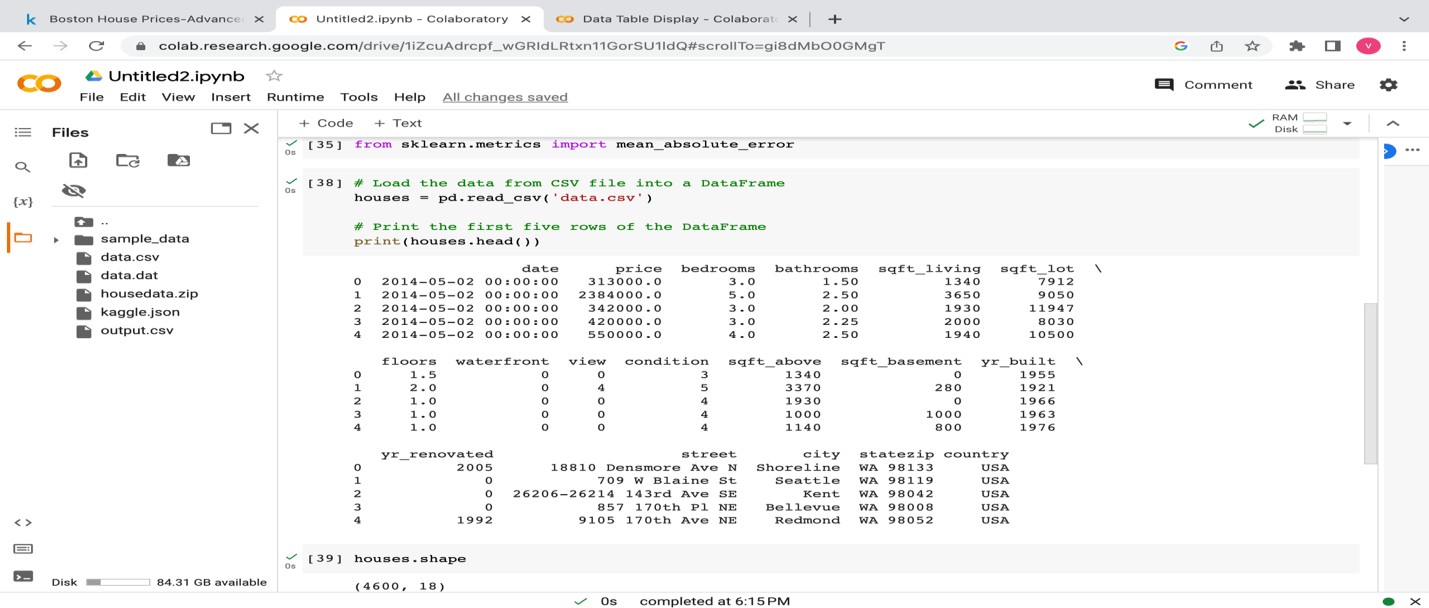
# Step 2: Building the model.

We implemented a model using Google Colab(which is jupyter notebook environment). The main key takeaways of the model development are:

1. Identifying the key features that affect house prices.
2. Model building.

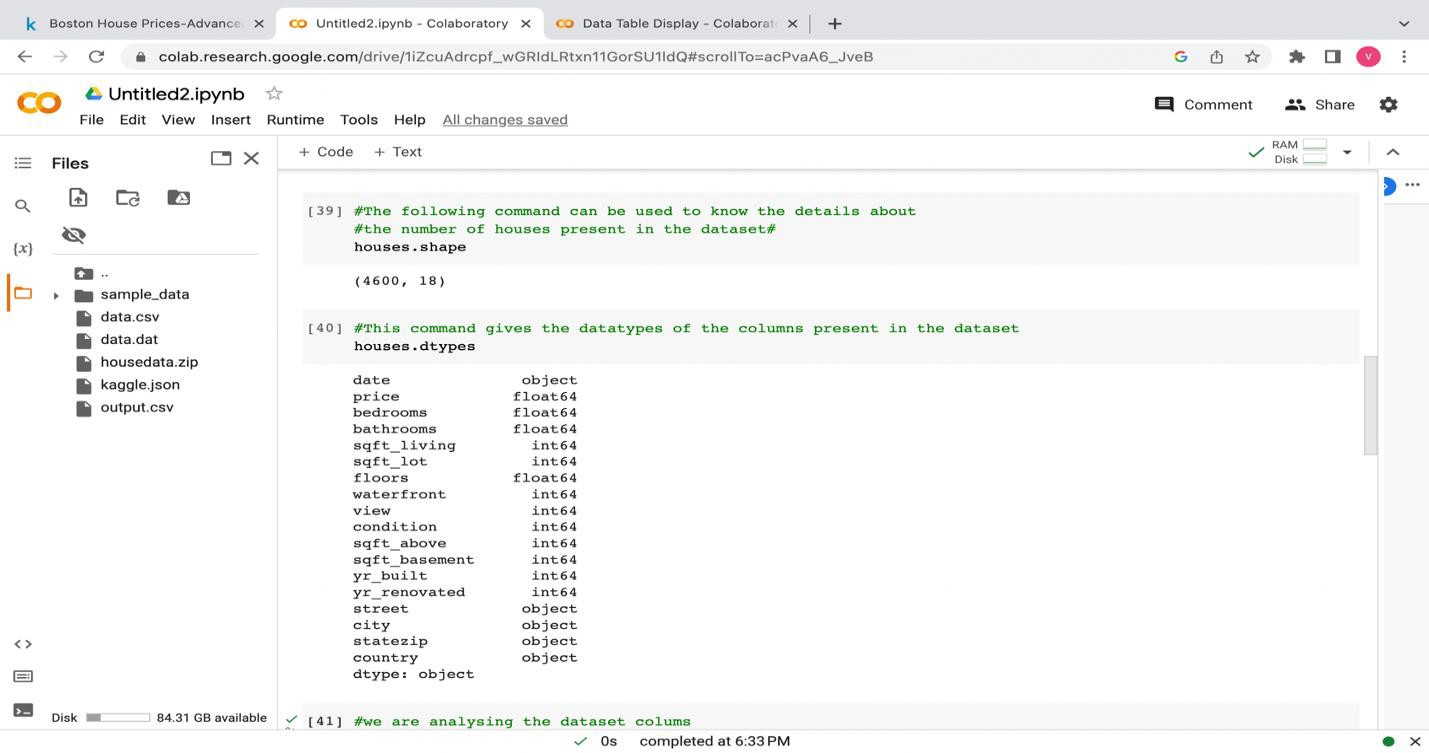
## Data Preparation.

The dataset should be imported into Google Drive and mounted on the notebook. Now import the required python packages and then load the dataset.

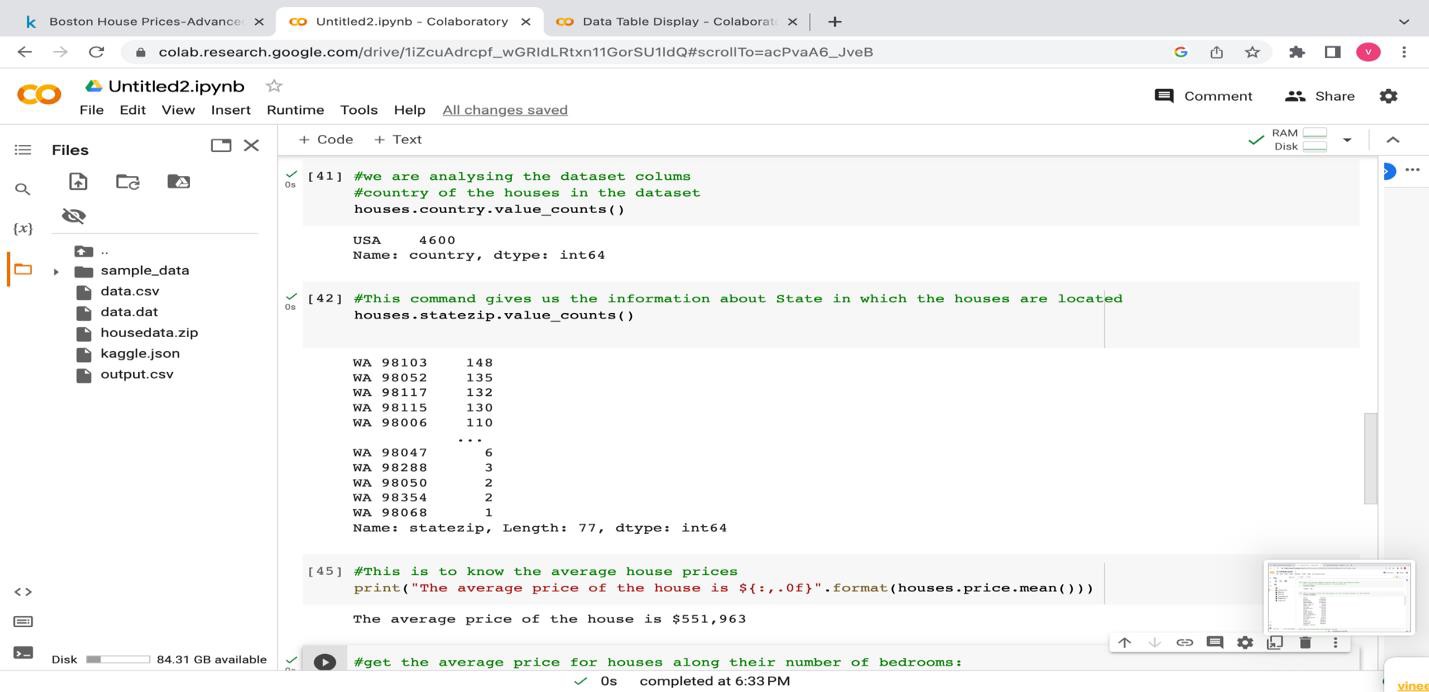


## Data Analysis:

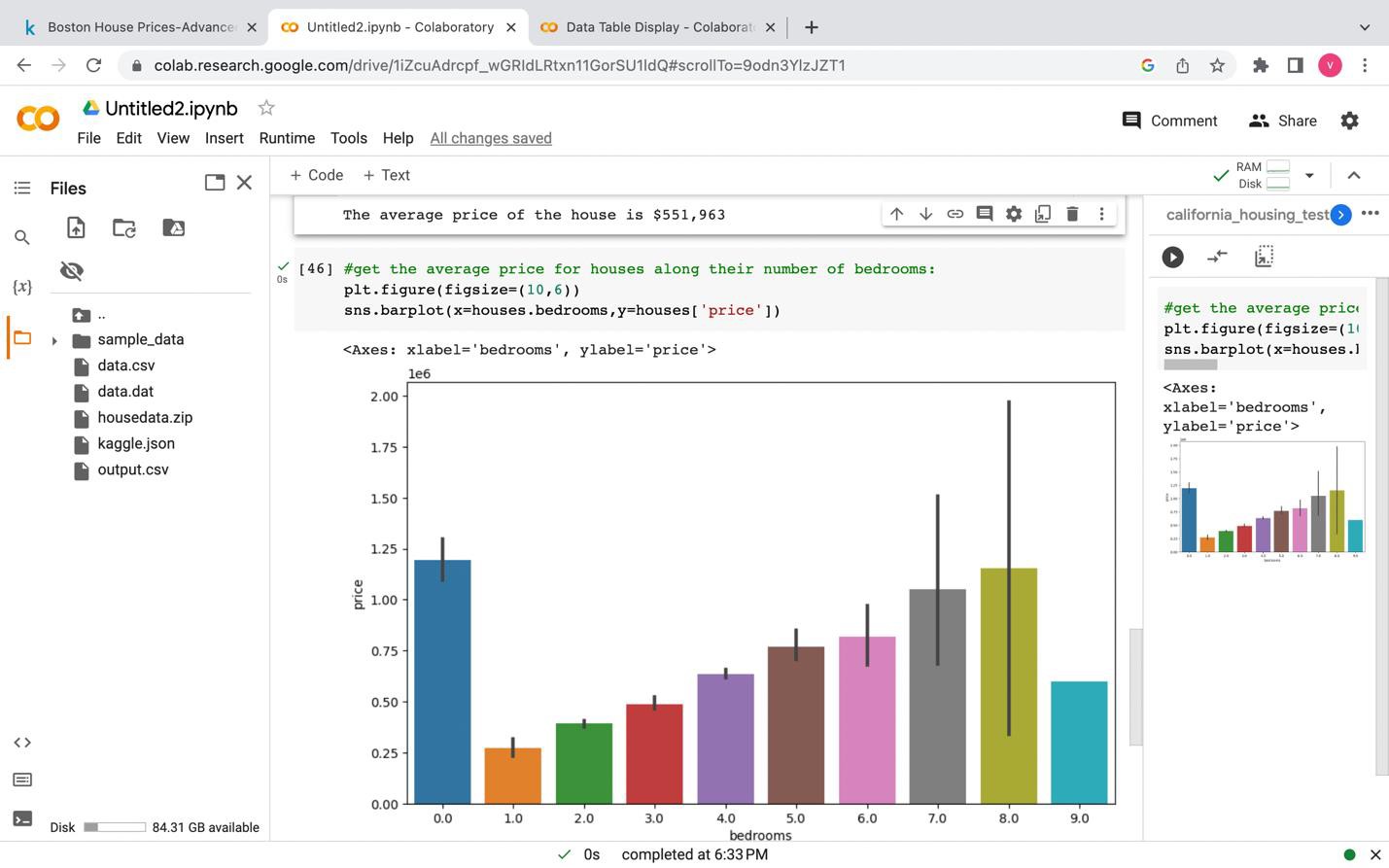
* 1. Determining the different types of data (categorical and numeric) that are included in the data set.



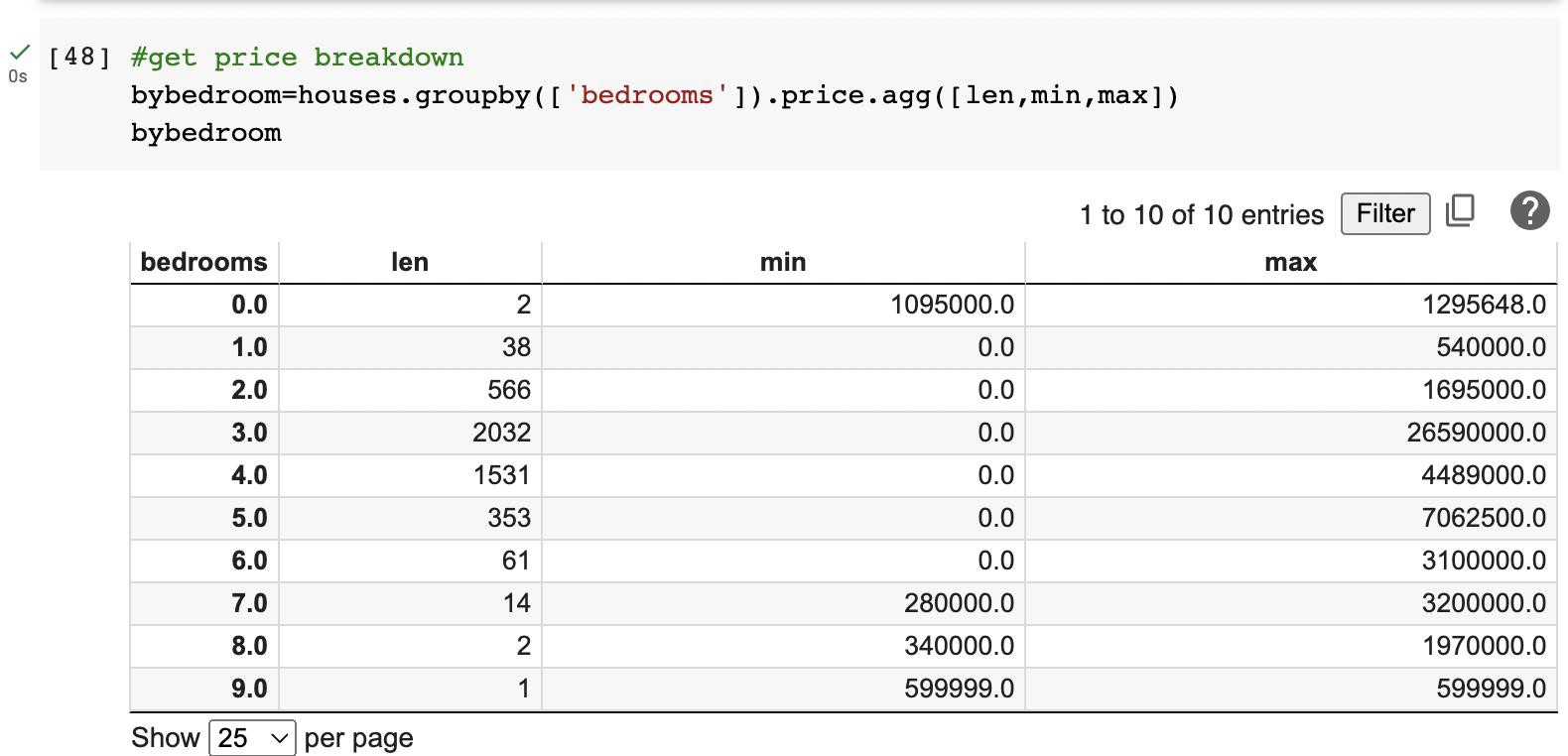
* 1. Here we are making some analysis like finding which country and state these houses are in and calculating the average price of the house.



* 1. Establishing the relationship between the number of bedrooms and the price of the house. Generally, the greater number of bedrooms the higher the price is. We created a data visualization using bar plot from matplotlib library in python.



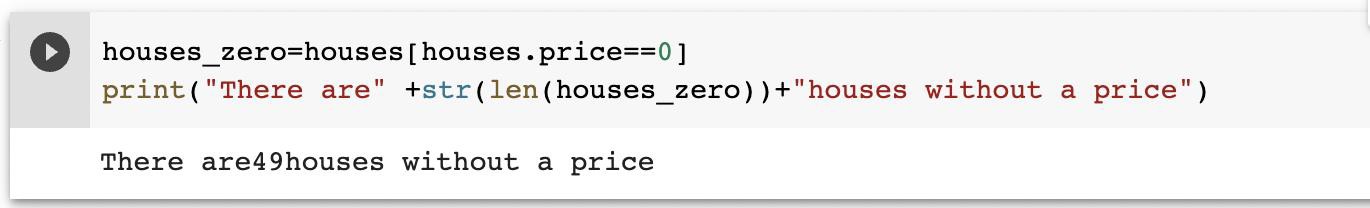
* 1. Making analysis of price breakdown for each bedroom group.



## Extrapolation from the previous action:

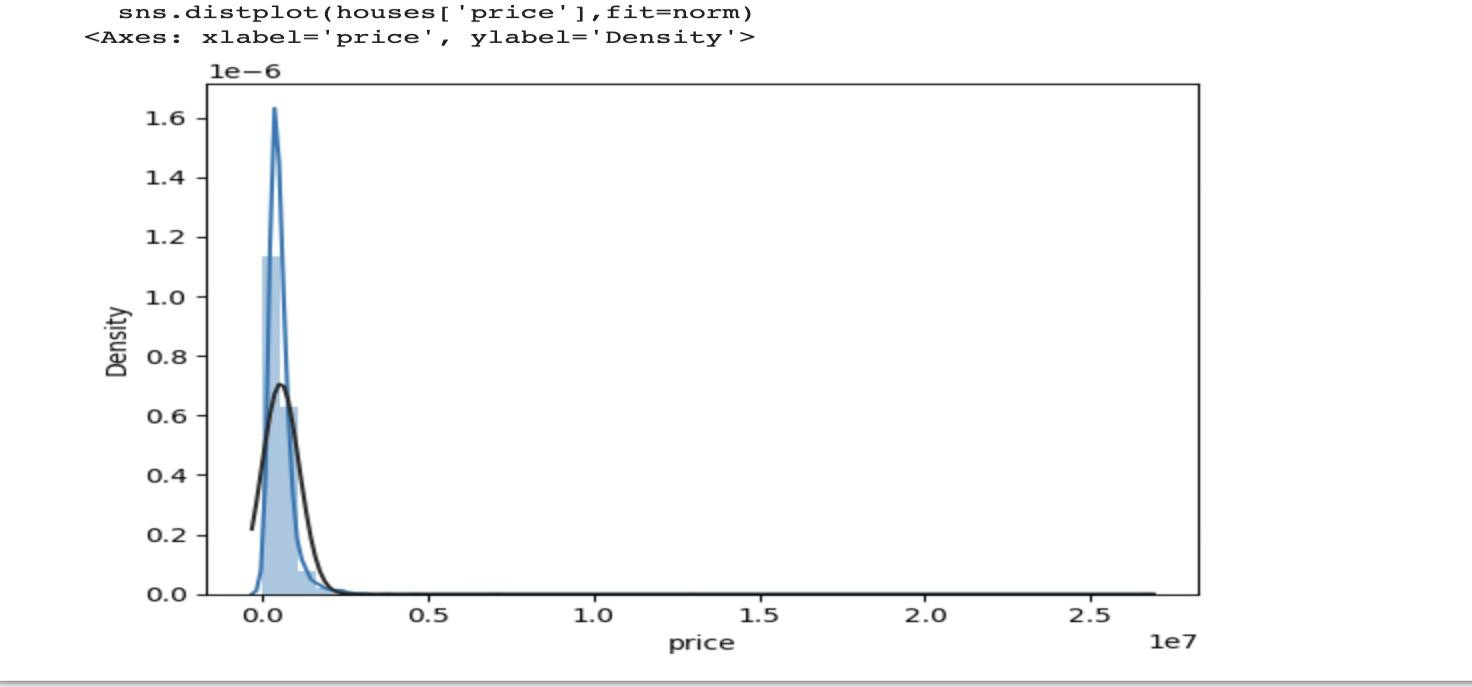
1. Some houses have a price of zero.
2. There are two houses with 0 bedrooms.
3. There is a max outlier in several bedrooms with 3.
4. There is only one house with 9 bedrooms and might be far away from the city.

## Let's address the issue of "houses with zero prices."



**Extrapolation from the previous action:**

* + Out of 4600 houses, there were 49 houses without price.



## Extrapolation from the previous action:

* + Blueline is the price distribution.
  + Blackline is the normal distribution.

## Addressing the following three problems my developing a new data frame.

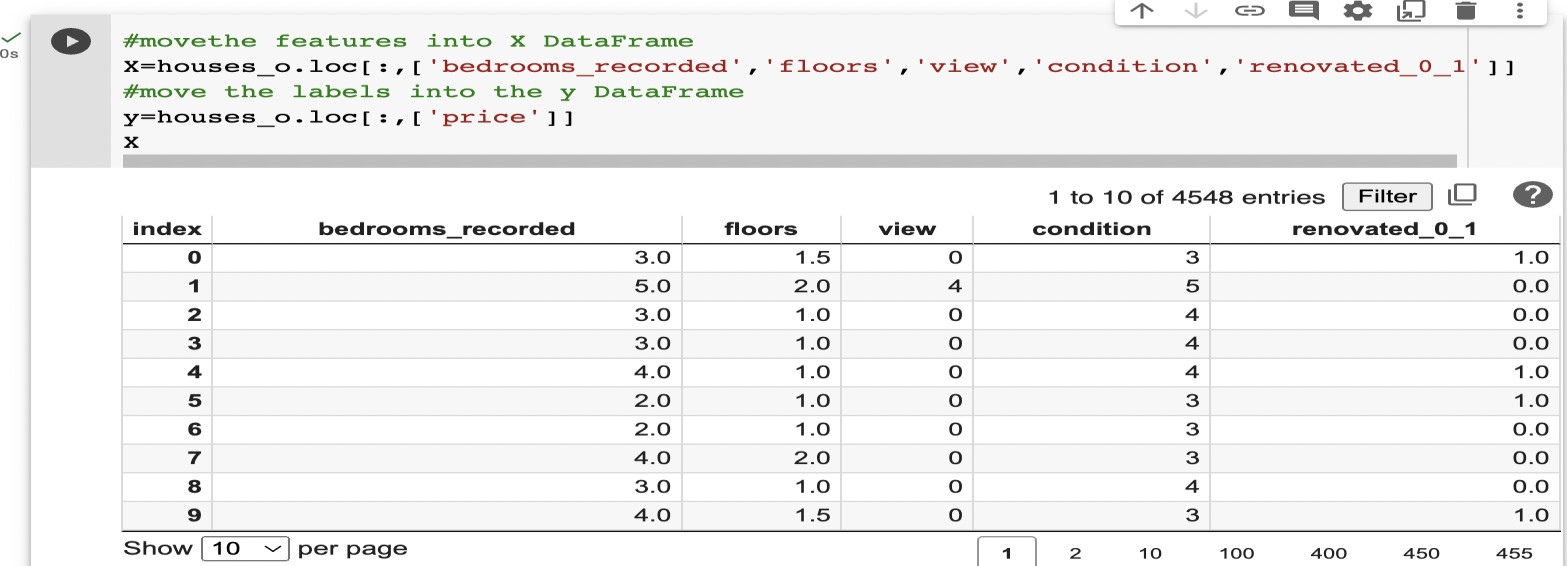
1. Houses with 0 bedrooms
2. Giant outlier at almost $27M — 50 times the price of a normal house
3. 49 Houses without a price

Following is the new data frame without problem 1,2 and 3.

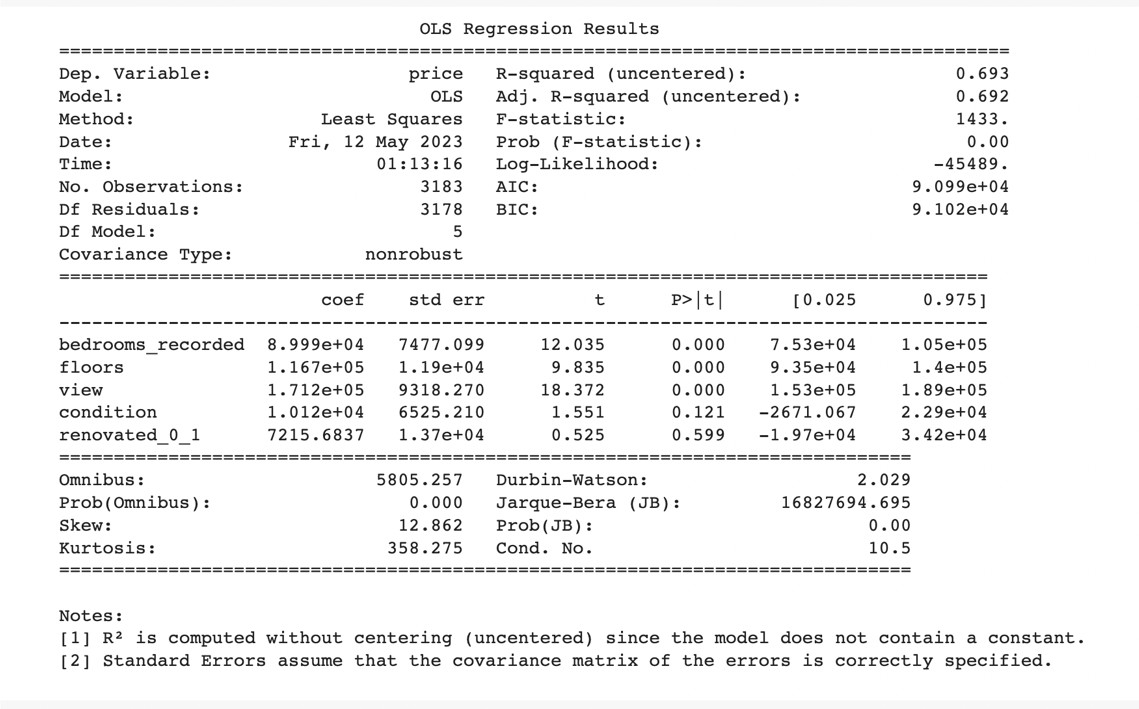
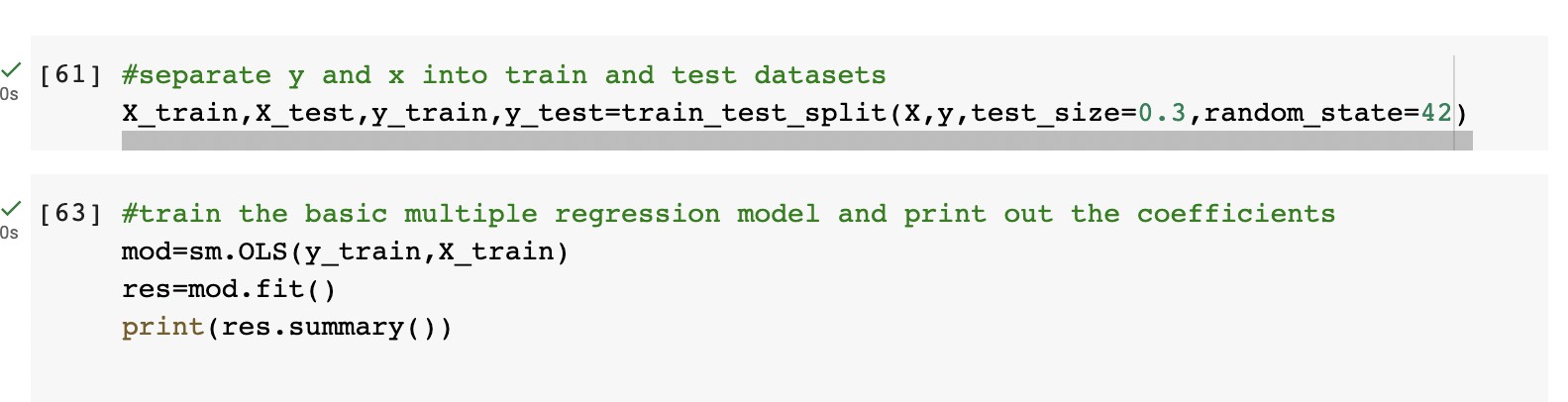


## Model Development:

Keeping the necessary features in the X data frame, and the predicted labels in the Y data frame.



Separated Y and X into train and test datasets. And then train the basic multiple regression model, that results in the coefficients.



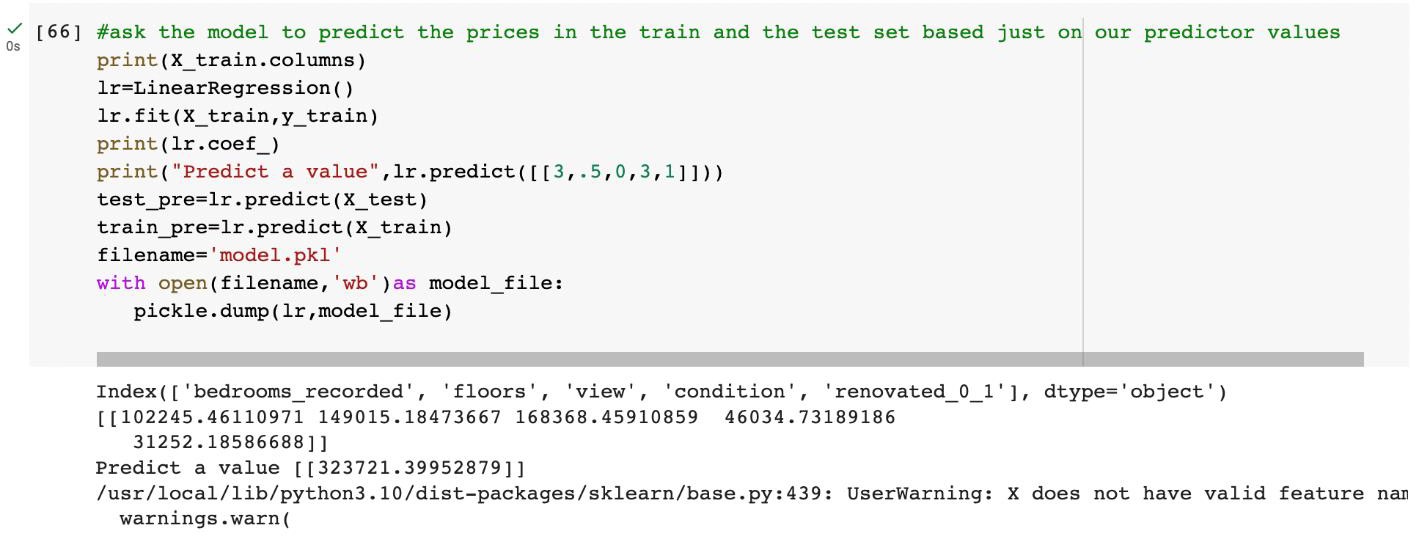
## Extrapolation from the previous action:

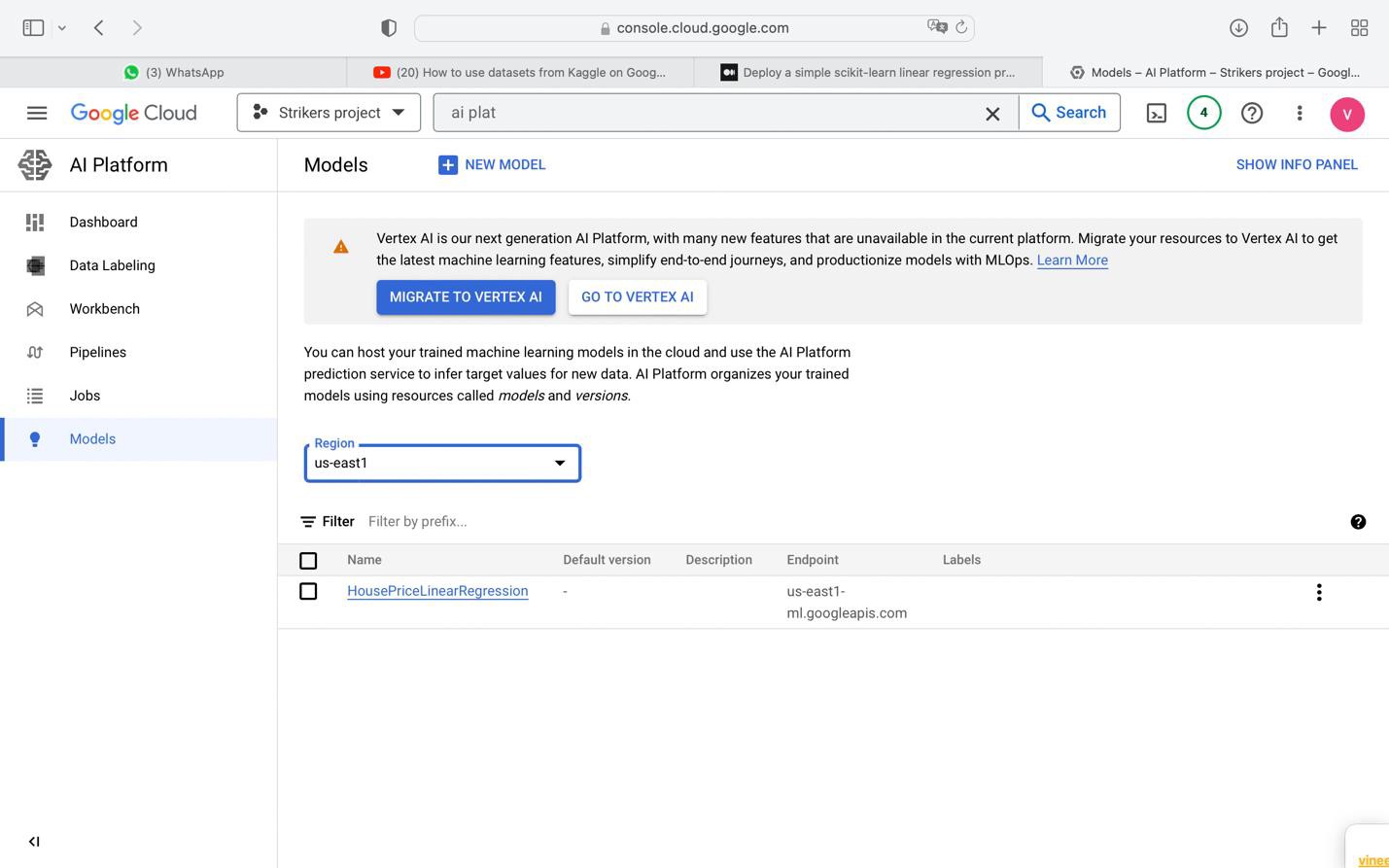
Based on the coefficients:

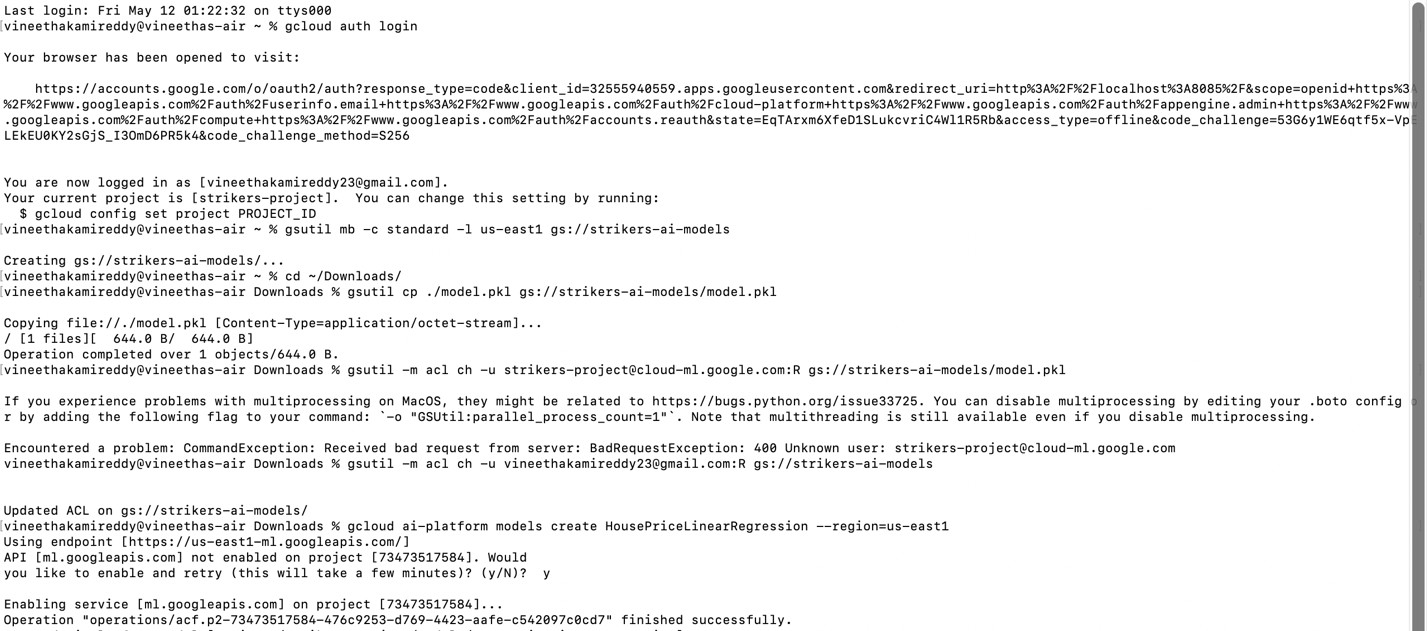
* There is a gain of $170k [1.712 x (10⁵)] for a single value change in view.

## Model Deployment:

Export the below developed model as a pkl file.

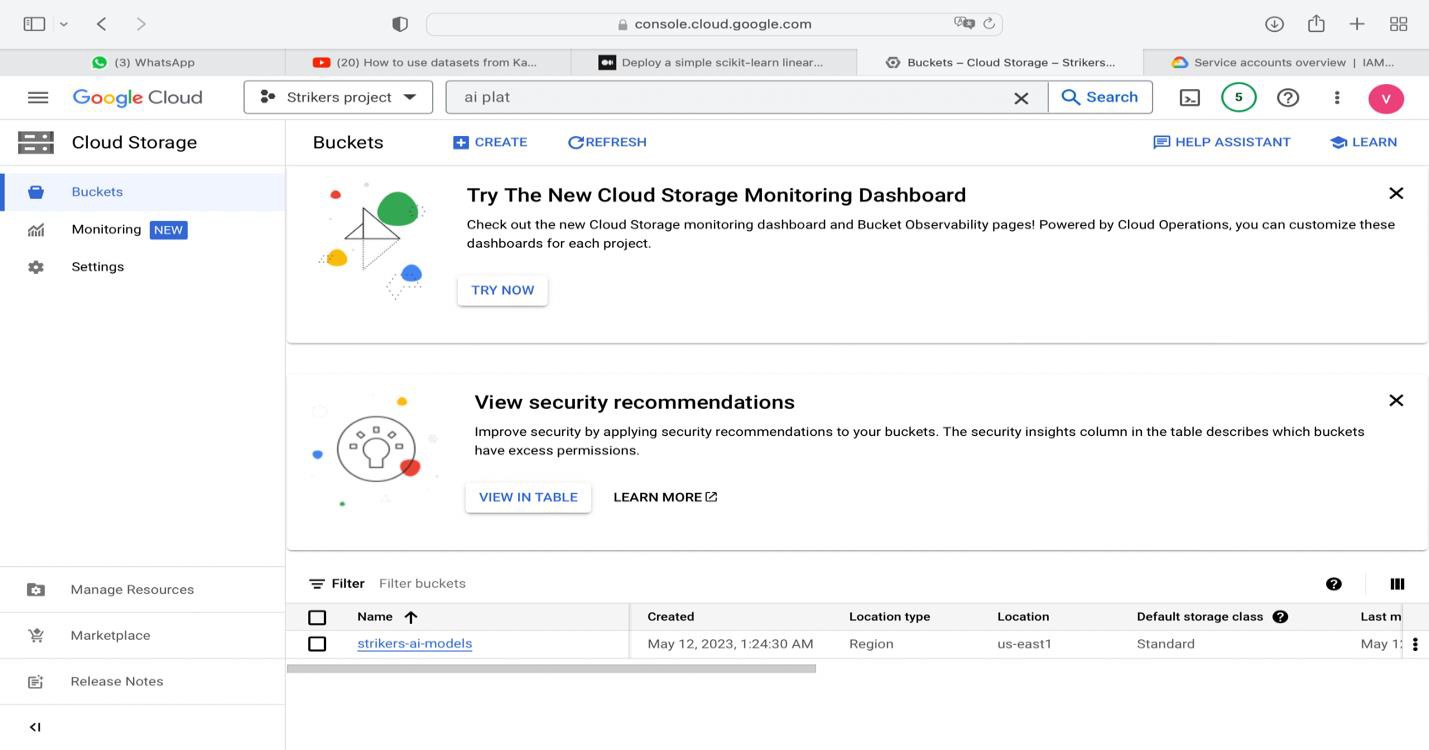






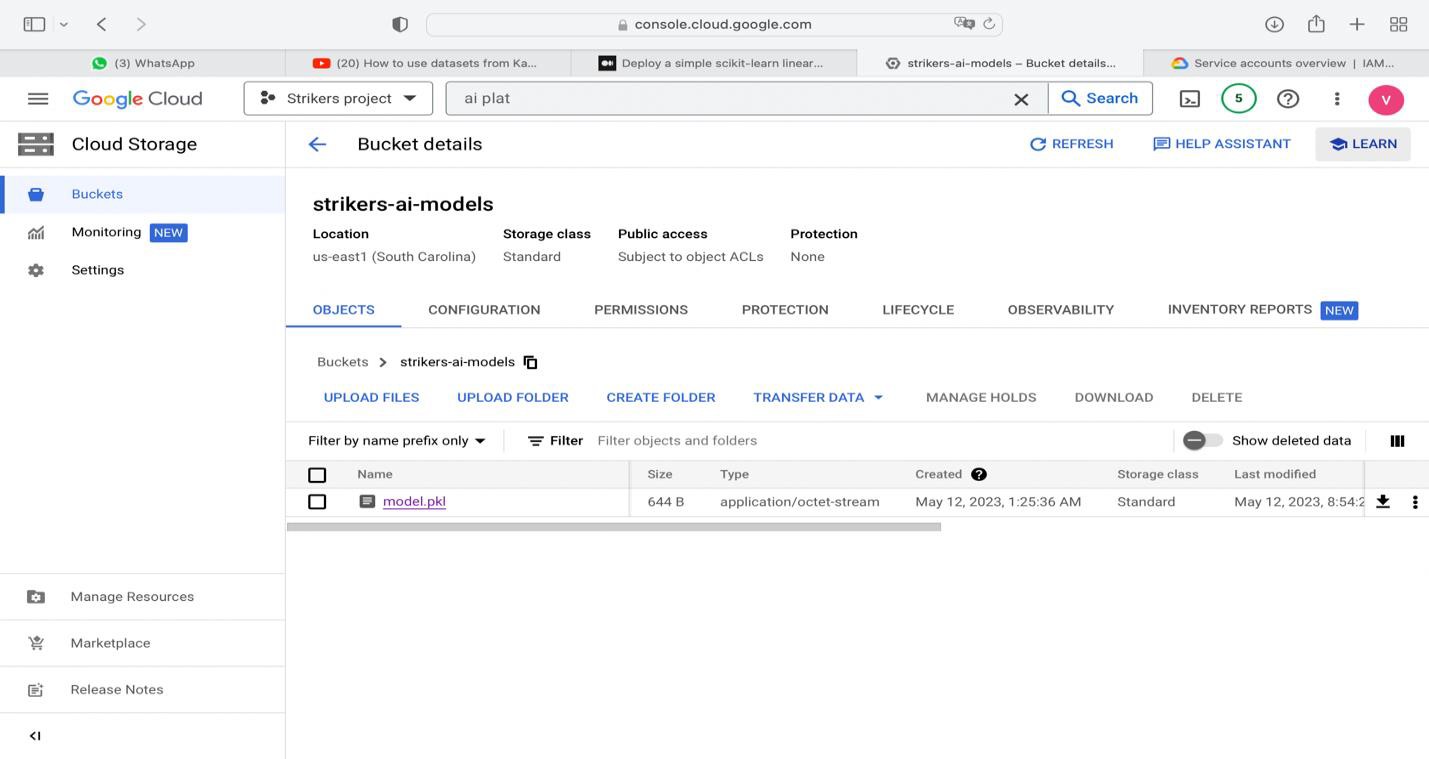
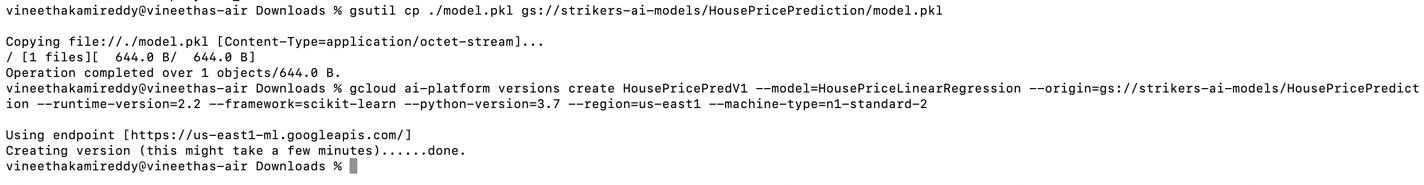
# Step3: Deploying model to the Google AI Platform.

## 1: Create and upload the model to Cloud Storage bucket.

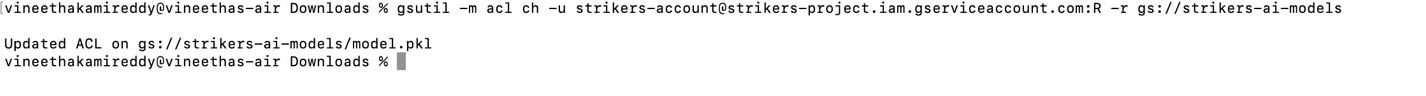


Note: Ensure the model file is in either pickle format [*model.pkl*] or joblib format [*model.joblib*].

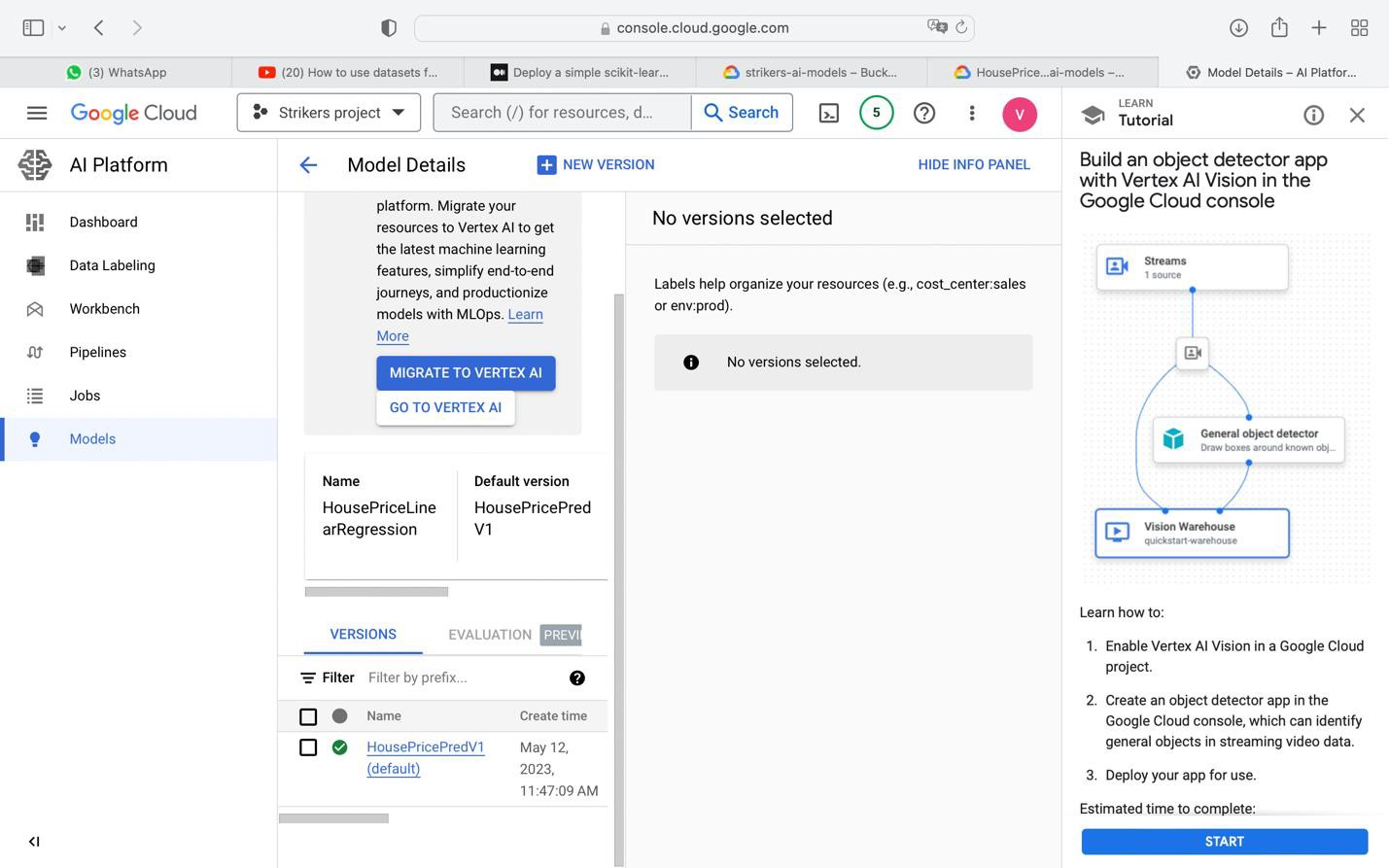
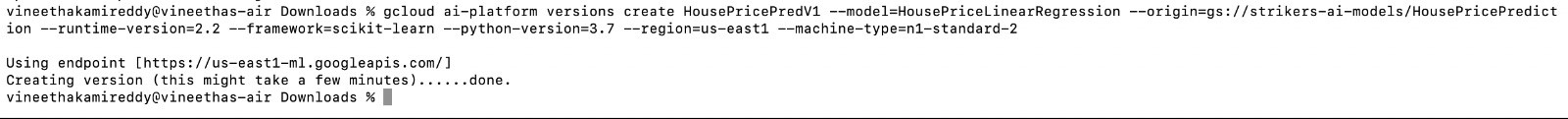
## 2: Navigate to the model downloaded folder and run the copy command.



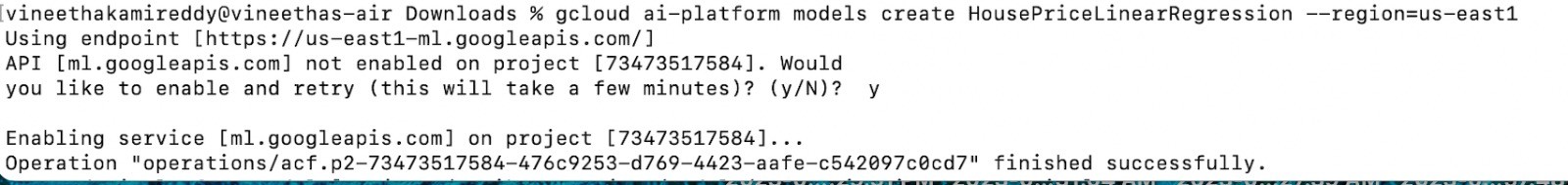
Note: Ensure the IAM service account has access to the storage bucket.

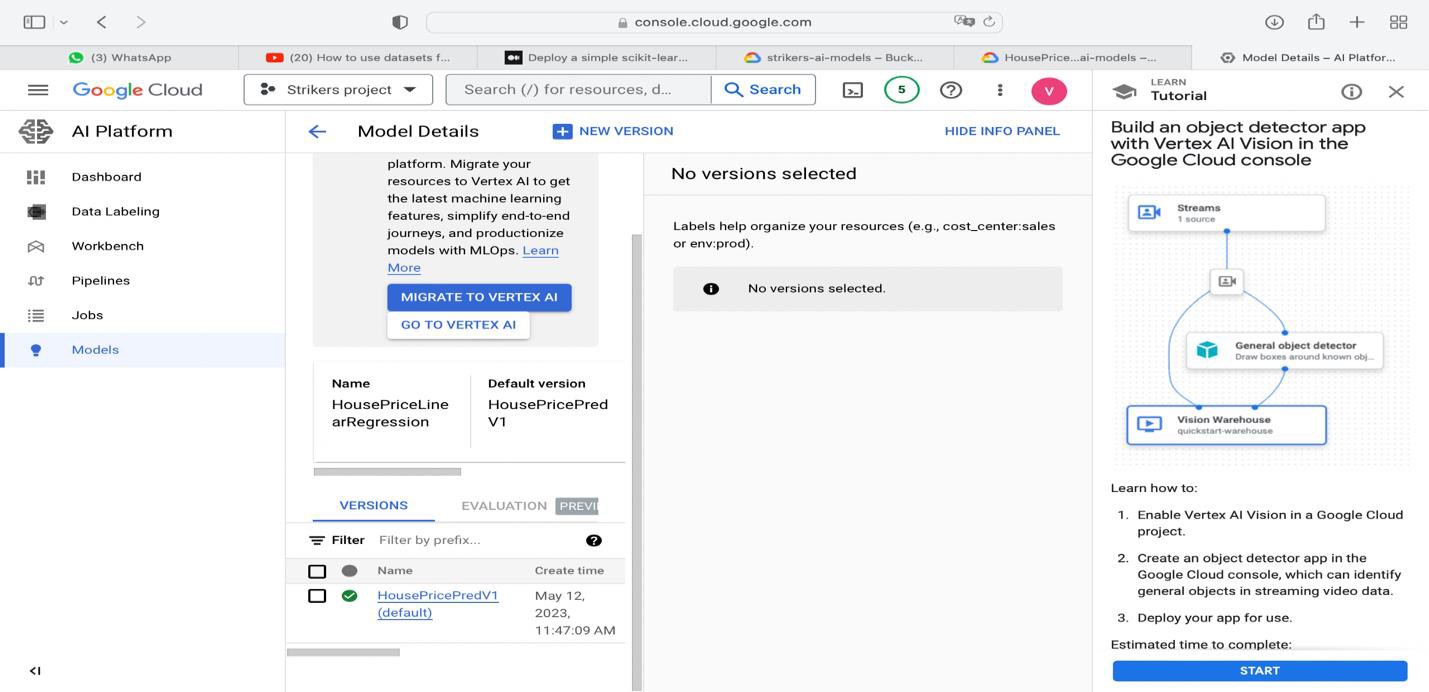


**3: Create the Model in the Google AI Platform.**



**4: Create the executable state of a model version.**





**5: Test the deployed predictive model.**

1. Click on the deployed model version and navigate to Test & Use tab.
2. Provide the input json data to the model.

