Lesson 2:
Introduction to Machine Learning

SEARCH

RESOURCES ▲

CONCEPTS ▼

✓ 17. Example One: House Price Pr...

✓ 18. Quiz: Example One

✓ 19. Example Two: Book Genre Exp...

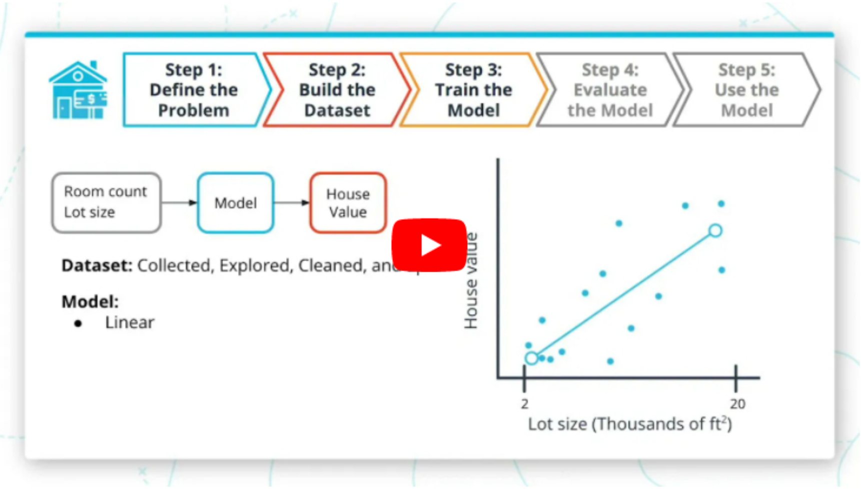
✓ 20. Quiz: Example Two

✓ 21. Example Three: Spill Detection...

✓ 22. Quiz: Example Three

✓ 23. Final Quiz

## Example One: House Price Prediction


Step 1: Define the Problem
Step 2: Build the Dataset
Step 3: Train the Model
Step 4: Evaluate the Model
Step 5: Use the Model
Room count
Lot size
Model
House Value
House value
2
20
Lot size (Thousands of ft²)

**Dataset:** Collected, Explored, Cleaned, an

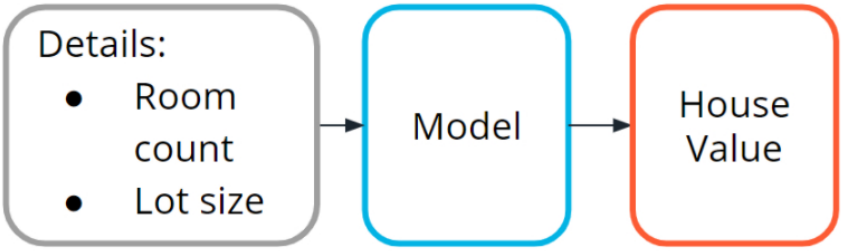
**Model:**

- Linear

House price prediction is one of the most common examples used to introduce machine learning.

Traditionally, real estate appraisers use many quantifiable details about a home (such as number of rooms, lot size, and year of construction) to help them estimate the value of a house.

You detect this relationship and believe that you could use machine learning to predict home prices.

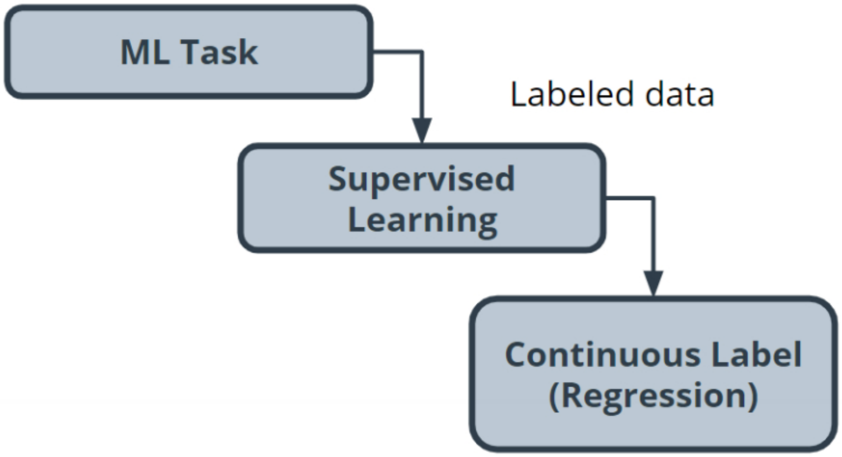


Machine language models to determine house values

### Step One: Define the Problem

> *Can we estimate the price of a house based on lot size or the number of bedrooms?*

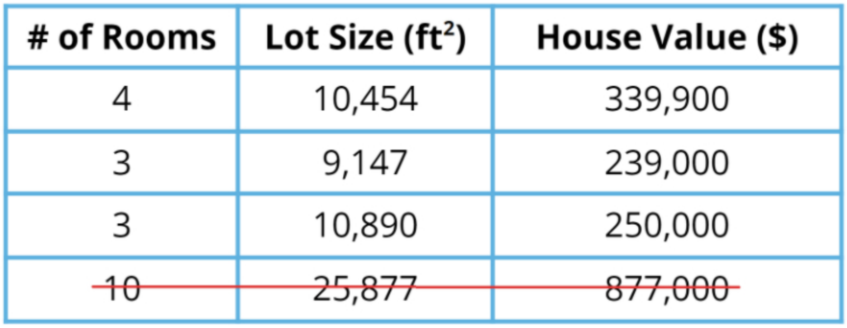
You access the sale prices for recently sold homes or have them appraised. Since you have this data, this is a *supervised learning* task. You want to predict a continuous numeric value, so this task is also a *regression* task.



Regression task

### Step Two: Building a Dataset

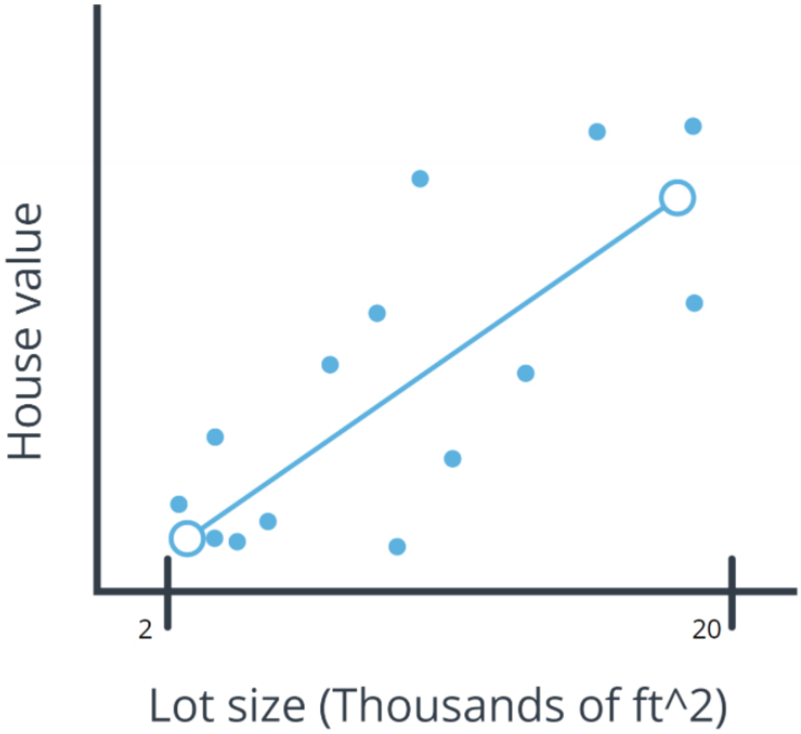
- **Data collection**: You collect numerous examples of homes sold in your neighborhood within the past year, and pay a real estate appraiser to appraise the homes whose selling price is not known.
- **Data exploration**: You confirm that all of your data is numerical because most machine learning models operate on sequences of numbers. If there is textual data, you need to transform it into numbers. You'll see this in the next example.
- **Data cleaning**: Look for things such as missing information or outliers, such as the 10-room mansion. Several techniques can be used to handle outliers, but you can also just remove those from your dataset.



| # of Rooms | Lot Size (ft$^2$) | House Value ($) |
|---|---|---|
| 4 | 10,454 | 339,900 |
| 3 | 9,147 | 239,000 |
| 3 | 10,890 | 250,000 |
| ~~10~~ | ~~25,877~~ | ~~877,000~~ |

Data cleaning: removing outlier values

- **Data visualization**: You can plot home values against each of your input variables to look for trends in your data. In the following chart, you see that when lot size increases, the house value increases.



Regression line of a model

### Step Three: Model Training

Prior to actually training your model, you need to split your data. The standard practice is to put 80% of your dataset into a training dataset and 20% into a test dataset.

### Linear model selection

As you see in the preceding chart, when lot size increases, home values increase too. This relationship is simple enough that a linear model can be used to represent this relationship.

A linear model across a single input variable can be represented as a line. It becomes a plane for two

A linear model across a single input variable can be represented as a line. It becomes a plane for two variables, and then a hyperplane for more than two variables. The intuition, as a line with a constant slope, doesn't change.

### Using a Python library

The Python `scikit-learn` library has tools that can handle the implementation of the model training algorithm for you.

### Step Four: Evaluation

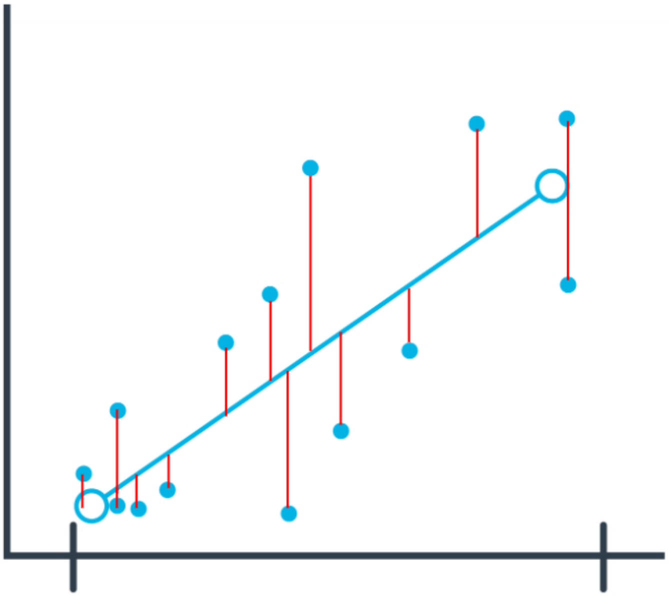
One of the most common evaluation metrics in a regression scenario is called *root mean square* or *RMS*. The math is beyond the scope of this lesson, but RMS can be thought of roughly as the "average error" across your test dataset, so you want this value to be low.

$$RMS = \sqrt{\frac{1}{n}\sum_{i} x_i^2}$$

The math behind RMS

In the following chart, you can see where the data points are in relation to the blue line. You want the data points to be as close to the "average" line as possible, which would mean less net error.

You compute the *root mean square* between your model's prediction for a data point in your test dataset and the true value from your data. This actual calculation is beyond the scope of this lesson, but it's good to understand the process at a high level.



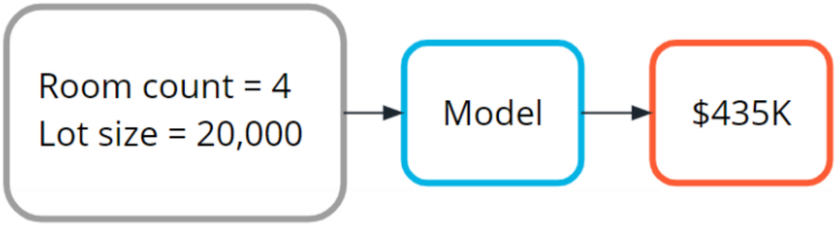
### Interpreting Results

In general, as your model improves, you see a better RMS result. You may still not be confident about whether the specific value you've computed is good or bad.

Many machine learning engineers manually count how many predictions were off by a threshold (for example, $50,000 in this house pricing problem) to help determine and verify the model's accuracy.

### Step Five: Inference: Try out your model

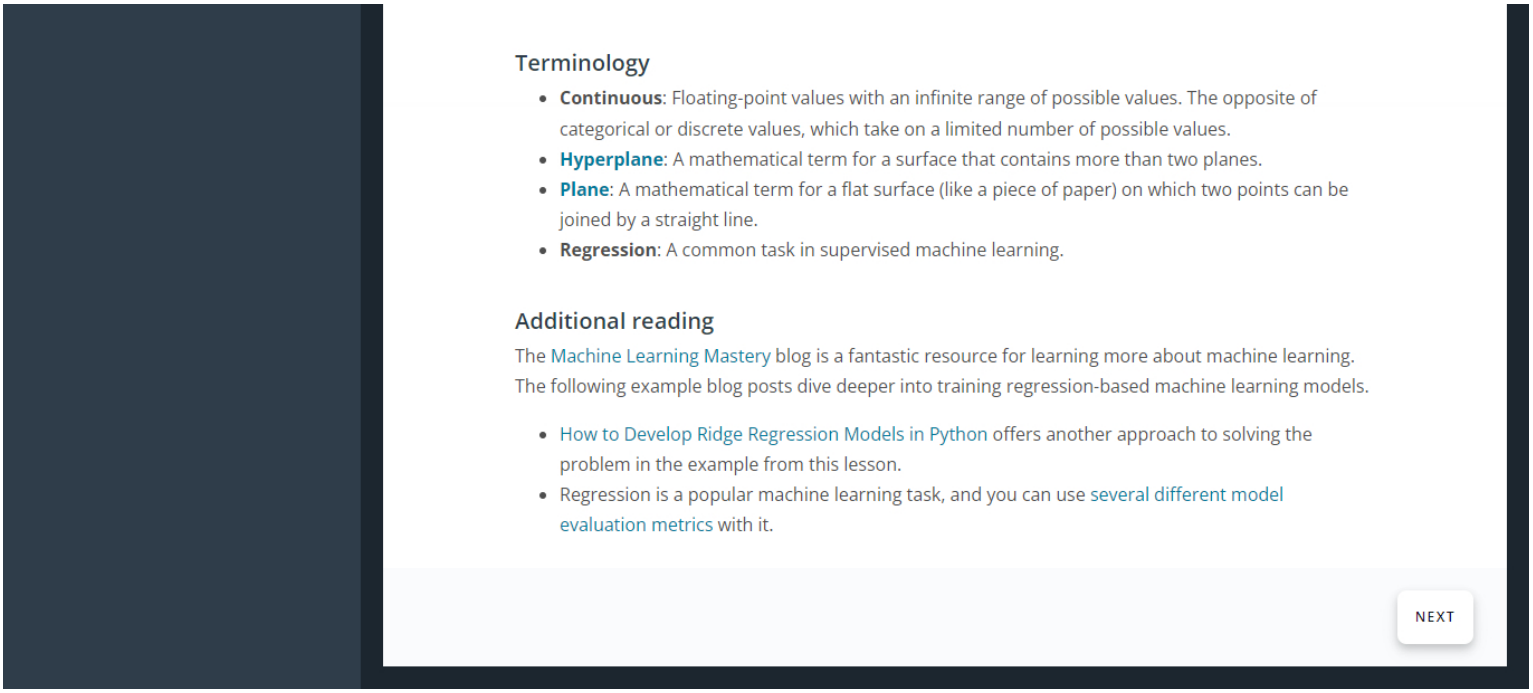
Now you are ready to put your model into action. As you can see in the following image, this means seeing how well it predicts with new data not seen during model training.



Room count = 4
Lot size = 20,000

Model

$435K



### Terminology

- **Continuous**: Floating-point values with an infinite range of possible values. The opposite of categorical or discrete values, which take on a limited number of possible values.
- **Hyperplane**: A mathematical term for a surface that contains more than two planes.
- **Plane**: A mathematical term for a flat surface (like a piece of paper) on which two points can be joined by a straight line.
- **Regression**: A common task in supervised machine learning.

### Additional reading

The Machine Learning Mastery blog is a fantastic resource for learning more about machine learning. The following example blog posts dive deeper into training regression-based machine learning models.

- How to Develop Ridge Regression Models in Python offers another approach to solving the problem in the example from this lesson.
- Regression is a popular machine learning task, and you can use several different model evaluation metrics with it.