

## Case Study: Regression

### PROJECT

## Case Study: Regression



### Scenario

You are working as an analyst for a real estate company. Your company wants to build a machine learning model to predict the selling prices of houses based on a variety of features on which the value of the house is evaluated.

### Objective

Your job is to build a model that will predict the price of a house based on features provided in the dataset. Senior management also wants to explore the characteristics of the houses using some business intelligence tools. One of those parameters includes understanding which factors are responsible for higher property value - \$650K and above.

### Dataset

← Go back    cessary files, as well as the dataset, can be found in the following repository:  
**[Mid-bootcamp project - Regression.](#)**

The dataset consists of information on 22,000 properties. The dataset consists of historic data of houses sold between *May 2014* to *May 2015*.

## Case Study: Regression

---

- **Id**: Unique identification number for the property
- **date**: the date the house was sold
- **price**: the price of the house
- **waterfront**: the house which has a view to a waterfront
- **condition**: How good the condition is (Overall). 1 indicates worn-out property and 5 excellent.
- **grade**: Overall grade given to the housing unit, based on the King County grading system. 1 poor, 13 excellent.
- **Sqft\_above**: square footage of house apart from the basement
- **Sqft\_living15**: Living room area in 2015(implies-- some renovations) This might or might not have affected the lotSize area.
- **Sqft\_lot15**: lotSize area in 2015(implies-- some renovations)

### Expected Outcomes

We encourage you to thoroughly understand your data and take the necessary steps to prepare your data for modeling before building exploratory or predictive models.

Since this is a classification model, you can use linear regression for classification for building a model. You are also encouraged to use other models in your project including KNN, decision trees for regression.

### 1. Explore the data

To **explore the data**, you can use the techniques that have been discussed in class. Some of them include using the describe method, checking null values, using Matplotlib and Seaborn for developing visualizations.

The data has many categorical and numerical variables. Explore the nature of data for these variables before you start with the **data cleaning** process and then **data pre-processing** (scaling numerical variables and encoding categorical variables).

### 2. Build a Model

Use different models to compare the accuracies and find the model that best fits your data. You can use the measures of accuracies that have been discussed in class. Please

## Case Study: Regression

### 3. Visualize

You will use **Tableau** to visually explore the data further. The questions have been provided [in this file](#).

#### Deliverables

1. **A slides deck:** use [slides.com](#) for simple yet effective templates.
2. **A presentation:** Your public presentation should last between 5 and 7 minutes.
3. **SQL Queries:** Access the [SQL Questions file](#) and make sure you run these queries using SQL.
4. **A Tableau Dashboard:** Follow the instructions in [this file](#) to build in Tableau.
5. **Python code:** Your code will be reviewed. Make sure you follow the best practices explained in the class so far.

#### Evaluation Criteria

- Critical Thinking
- Communication
- Transparency
- Thoroughness

### Rubrics

[These are the rubrics](#) we use to assess your work.

#### Tips & Tricks

- Organize yourself (don't get lost!). Respect deadlines.
- Ask for help but don't forget that Google is your friend.
- Define a simple approach first. You never know how the data can betray you. 😊
- Document your work.
- Learn about the problem and what research has been done before you.
- Before making a graph, think about what you want to represent.

## Case Study: Regression

---

Mark as completed

PREVIOUS



Lab | Random variable  
distributions

NEXT

Case Study:  
Classification