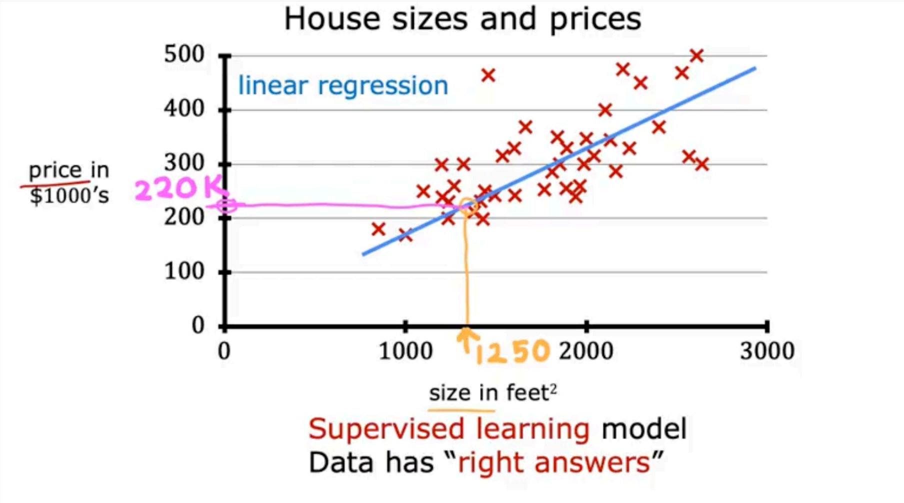
Regression Model

Linear Regression

Regression model predicts numbers.

💡 Examples: House sizes and prices



- Convert to table:

Size in $feet^2$	price in $1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

Notation:

$x =$ "input" variable feature
$y =$ "output"/"target" variable feature
$m =$ number of training example
$(x, y) =$ single training example
$(x^{(i)}, y^{(i)}) = i^{th}$ training example $(1^{st}, 2^{nd}, 3^{rd}, ..)$

Data Representation

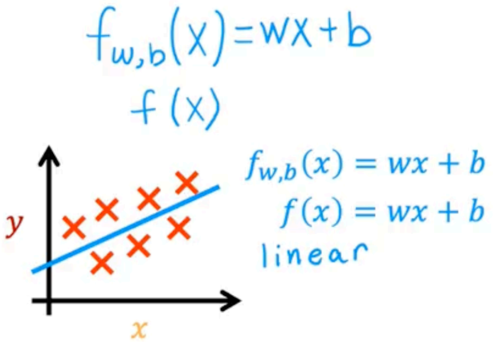
- The dataset used for training consists of input features (house sizes) and output variables (house prices), organized in a table format.
- Standard notation is introduced for describing the dataset, including input variables (x), output variables (y), and the total number of training examples (m).

Function Representation and Prediction

- The function f takes an input x and produces an estimate or prediction, represented as y-hat (ŷ).
- The model's prediction (ŷ) may differ from the actual target value (y) in the training set.

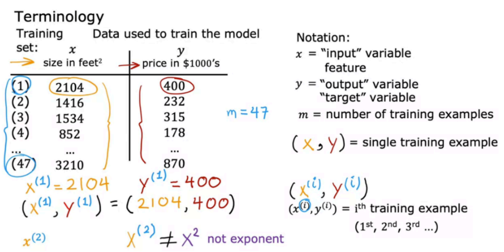
Linear Regression Model

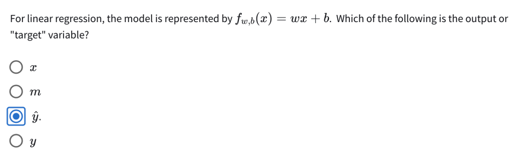
- The function f is often represented as a linear equation: f_w,b(x) = wx + b, where w and b are parameters that influence predictions.



> 💡 This is ***linear regression with one variable***, Another name is ***univariate linear regression.*** **Univariate** is just a fancy way of saying one variable.

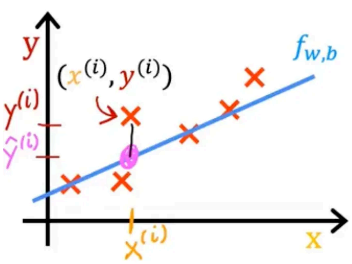
- Linear regression is a foundational model that simplifies the prediction process, making it easier to understand before moving to more complex models.




For linear regression, the model is represented by $f_{w,b}(x) = wx + b$. Which of the following is the output or "target" variable?

- ○ x
- ○ m
- ◉ $\hat{y}$.
- ○ y

Cost function



$$\hat{y}^{(i)} = f_{w,b}\left(x^{(i)}\right) = wx^{(i)} + b$$

Squared error cost function

$$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}\left(\hat{y}^{(i)} - y^{(i)}\right)^2 \longrightarrow error$$

$m =$ number of training examples.

By convention, the cost function that machine learning people use actually divides by 2 times m. The extra division by 2 is just meant to make some of our later calculations look neater, but the cost function still works whether you include this division by 2 or not.

💡 With knowledge about **SSE** learned from "Xác suất thống kê".

⇒ COST FUNCTION is the **DIVISION** of **SSE** with 2m

After replace f function:

$$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{w,b}\left(x^{(i)}\right) - y^{(i)}\right)^2 \longrightarrow error$$

💡 **Purpose:** Find w, b that $\hat{y}^{(i)}$ is close to $y^{(i)}$ for all $(x^{(i)}, y^{(i)})$

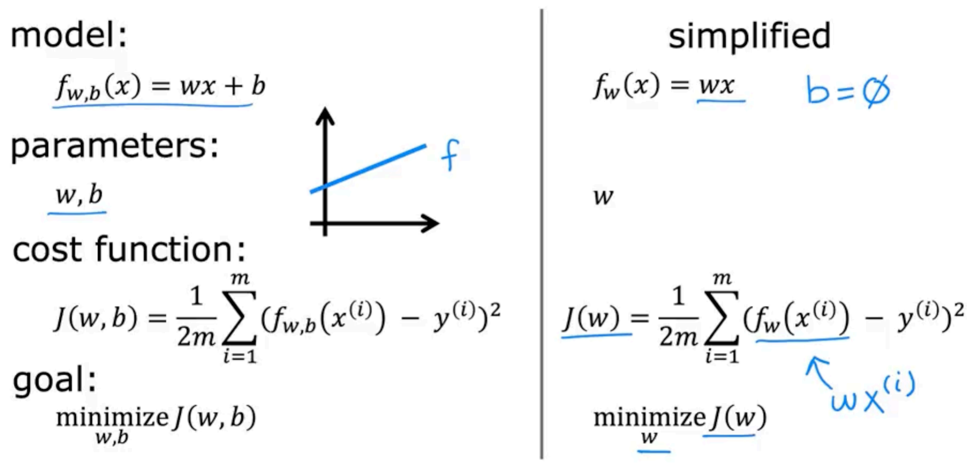
During training, the model adjusts its **bias** (b) and **weights** (w) to *minimize the **difference*** between predicted and actual values.

💡

- A cost function J measures the difference between predicted values and actual values in a linear regression model.
- The goal is to minimize J by adjusting the model parameters w and b to achieve the best fit for the training data.

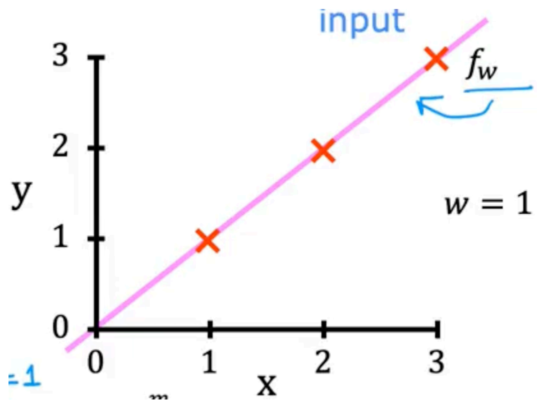
Simplified Model Analysis (for easier analysis)

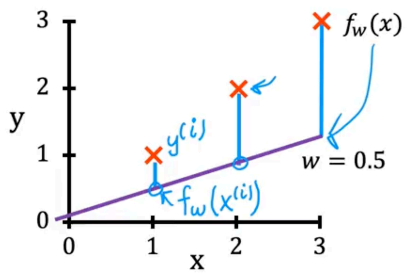
- Simplify: $b = 0$ ⇒ only w to optimize. ⇒ simplify variable for solving PROBLEM: **MINIMIZE COST FUNCTION**.

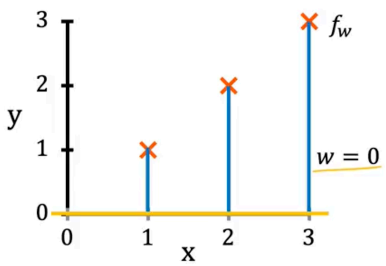


- The relationship between the model function $f_w(x)$ and the cost function $J(w)$ is visualized through graphs.

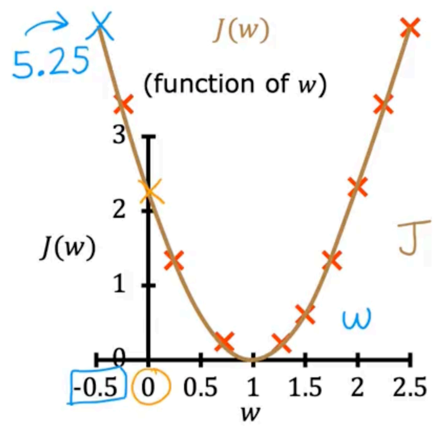
- With fixed w, function of $f_w(x)$:







- With each value of w, using the formula of $J(w)$ ⇒ We can describe $J(w)$ as a graph. (the following one is quadratic graph).



Parameter Optimization

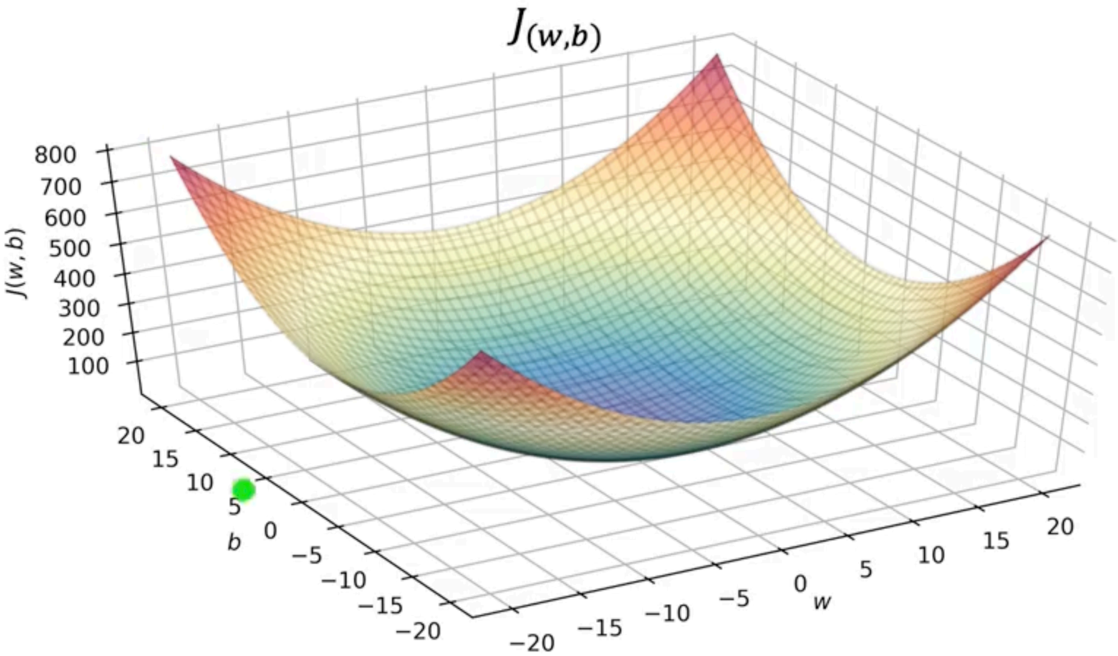
- Different values of w lead to different straight line fits, affecting the cost J.
- The optimal value of w is identified as the one that minimizes the cost function, resulting in the best fit for the training data.

❓ **Ques: When does the model fit the data relatively well, compared to other choices for parameter w?**

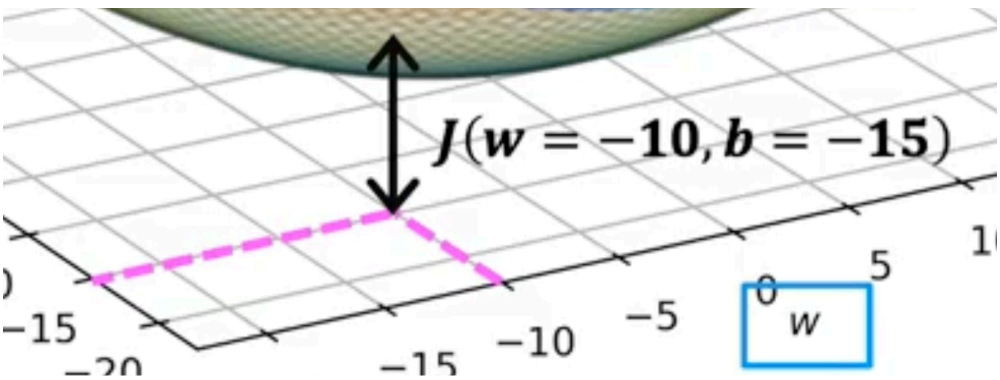
⇒ When the cost J is at or near a minimum.

Cost Function Visualization

- With **2 variables**: w, b ⇒ To visualize $J(w, b)$, we need to graph it on $3D$ space. The following is a typical instance:



- In a position with $w = -10, b = -15$, we have $J(w, b)$:

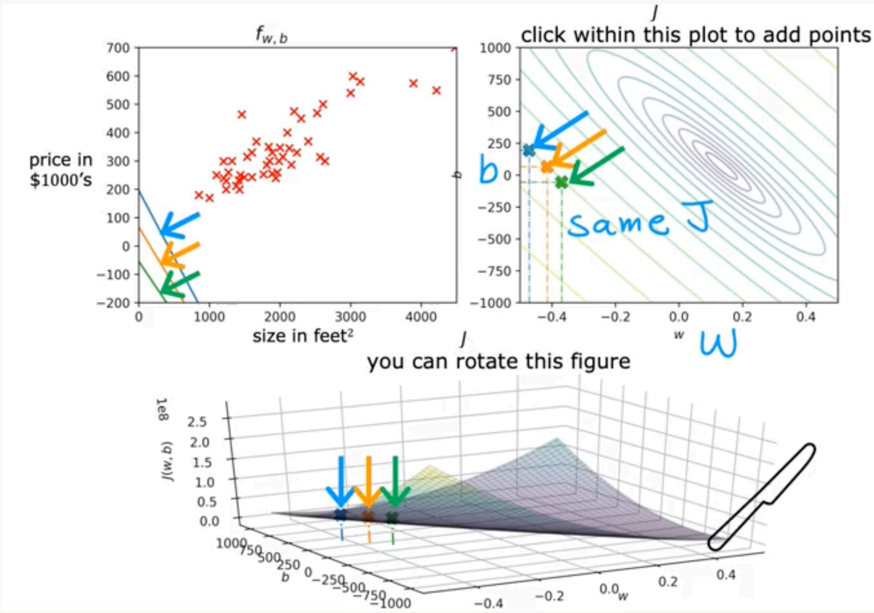


💡 **Contour plots** provide a 2D representation, showing levels of equal cost function values, helping to visualize the relationship between parameters.

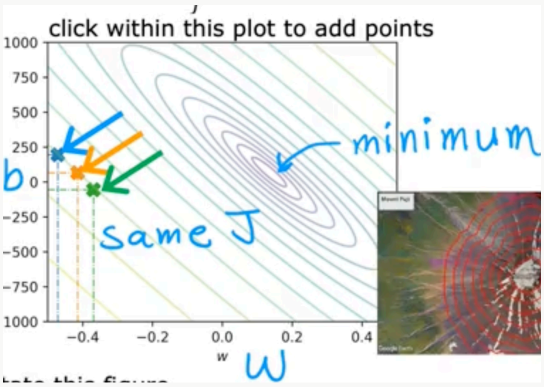
Contour plots show levels of a third variable (like the cost function J) on a 2D plane using lines that connect points with equal values.

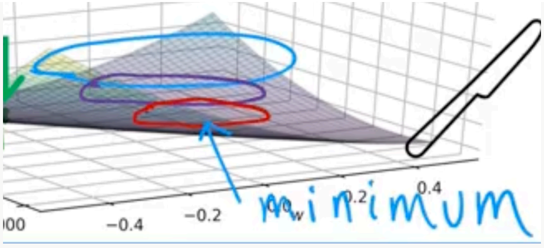
- **Horizontal axis**: one parameter (e.g., w)
- **Vertical axis**: another parameter (e.g., b)
- **Contour lines**: show where J is constant across combinations of w and b

Like using a knife to cut a slice. The following describe corresponding points in each plot graph.



*The **minimum point** of the cost function, where the model performs best, is located at **the center of the contour plot's smallest oval.***





More about contour plot:

https://www.youtube.com/watch?v=WsZj5Rb6do8&t=149s

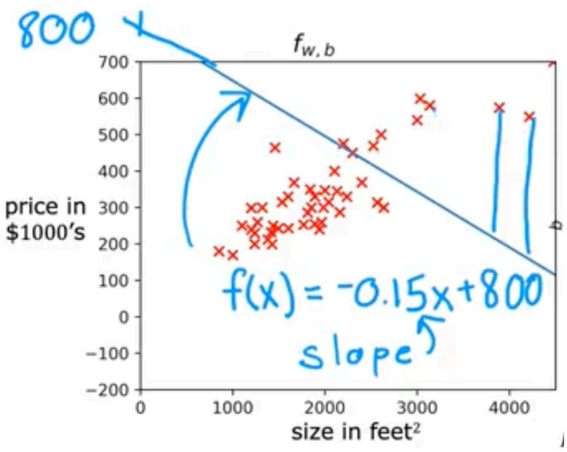
https://www.youtube.com/shorts/i-I7DxTwUTs

Visualization examples:

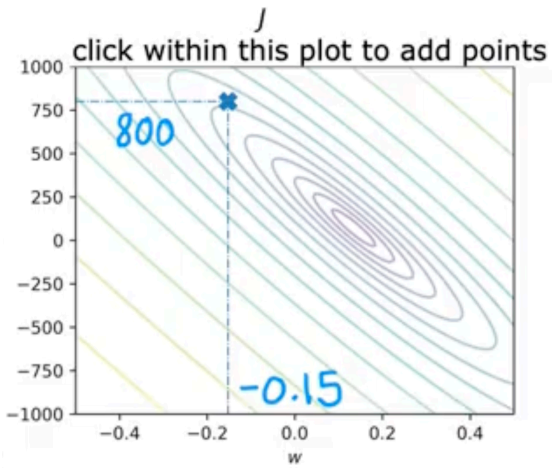
Explains how parameters w and b in linear regression **affect** the cost function J.

- **Visualization**: A graph shows how different w and b values impact J. For example, $w = -0.15, b = 800$ results in a high cost (poor fit).

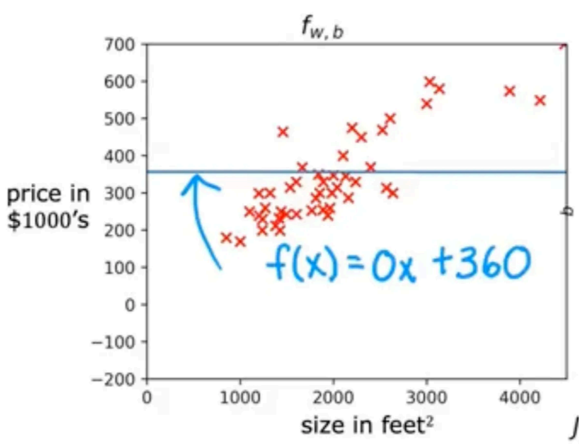
When we choose $(w, b) = (-0,15; 800)$



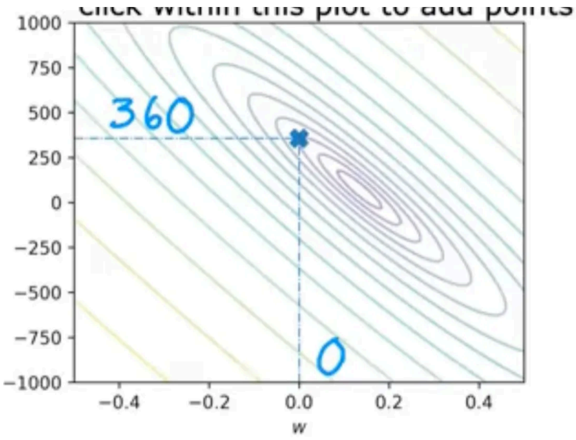
Then the position of this point is far from the minimum of contour plot graph.



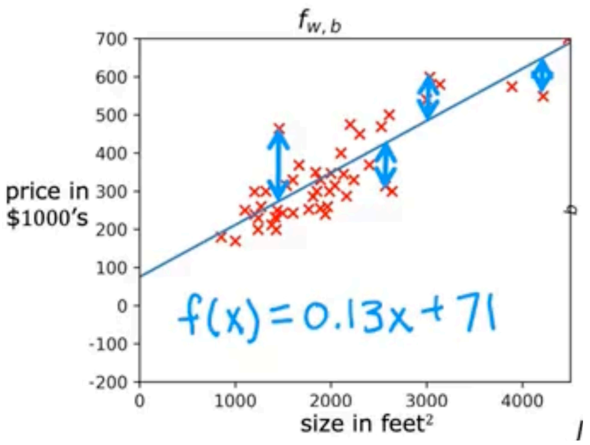
When we choose $(w, b) = (0; 360)$



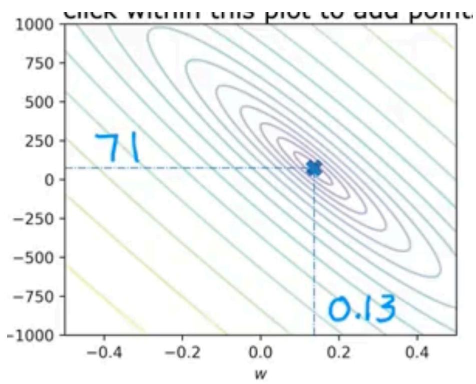
Then the position of this point is nearer from the minimum of contour plot graph.



When we choose $(w, b) = (0.13; 71)$



Then the position of this point is at the minimum of contour plot graph. ⇒ ***good fit***



- **Fit Quality**: Lower cost means better fit; the goal is to minimize J.
- **Gradient Descent**: Introduced as an efficient method to ***find optimal*** w ***and*** b. It's key to training many ML models and will be covered further next.

💡 **Practice Ques:**

For linear regression, the model is $f_{w,b}(x) = wx + b$.

Which of the following are the inputs, or features, that are fed into the model and with which the model is expected to make a prediction?

- ○ w and b.
- ◉ x
- ○ (x, y)
- ○ m

For linear regression, if you find parameters w and b so that $J(w, b)$ is very close to zero, what can you conclude?

- ◉ The selected values of the parameters w and b cause the algorithm to fit the training set really well.
- ○ The selected values of the parameters w and b cause the algorithm to fit the training set really poorly.
- ○ This is never possible -- there must be a bug in the code.