✏️

Regression Model

Course	
Topic	
Type	Lecture
Done	☑
Date	@July 11, 2022
Materials	https://drive.google.com/file/d/1EQC2CnTYN7EUieKc_0v7RkATU1IZGDkO/view?usp=sharing

▼ Summary

Linear Regression Model

Common terminology used in machine learning

Cost Function

Cost Function Intuition

Visualizing the cost function

Linear Regression Model

- The first model: Linear Regression (fitting a straight line to data)
- Ex. Predicting a price based on the size of a house
- This is supervised learning since we train with the price and size.
- Specifically the type of supervised learning is a regression and not a classification
- The data table might look like:

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

Each row in the table is one data point

Figure 1: Portland Dataset with house size and price

Common Terminology used in machine leraning:

Training Set:

- Data used to train the model

row #	x: size in feet ^{2}	y: price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

Figure 2: Explaning different types of variable in dataset

💡 x: "input" variable / feature

> 💡 y: "output" variable / "target" variable

> 💡 m: number of training examples

> 💡 (x,y): one training example

> 💡 (x^i, y^i): ith training example

> 💡 e.g. $x^1 = 2104$ and $y^1 = 400$ $(x^1, y^1) = (2104, 400)$

- **Training set has:**
    - **"input features" - x**
    - **"output features" - y**
- Process of generating a supervised learning model: (see Fig 3)
    1. Training set (features and targets)
    2. Learning algorithm
    3. x → f → $\hat{y}$ where the above values correspond with:
    - **x: feature**
    - **f: hypothesis**
    - **$\hat{y}$: prediction**
- The f hyothesis function:

    > 💡 $f_{w,b}(x) = wx + b$

    > 💡 One variable linear regression

💡 Univariate linear regression

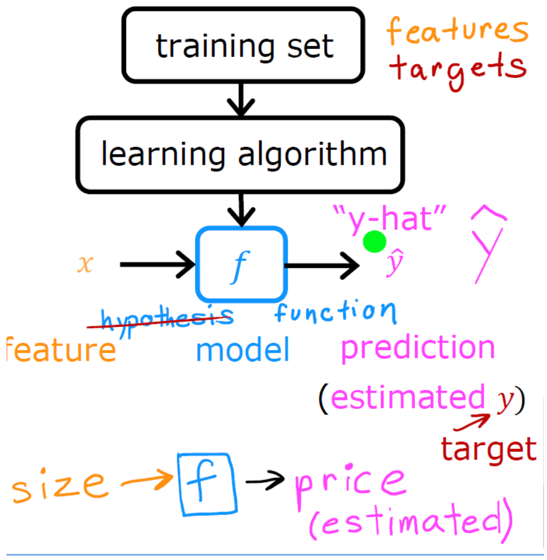

Figure 3: Process of generating a supervised learning model

Cost function:

💡 Linear Regression Model: $f_{w,b}(x) = wx + b$

- w, b: ***parameters/coefficients***. This is what we need to edit when training the model. These variables are like the slope and y-intercept respectively

Determining squared error cost function

- Error: $\hat{y}^{(i)} - y^{(i)}$
- Cost function: $J(w,b) = \frac{\sum_{i=1}^{m}(\hat{y}^{(i)} - y^{(i)})^2}{2m}$

- NOTE: It is divided by $2m$ instead of m to make later calculations easier

- Expanding $\hat{y}^{(i)}$: $J(w,b) = \frac{\sum_{i=1}^{m}(f_{w,b}(x^{(i)}) - y^{(i)})^2}{2m}$
- The cost function used for linear regression is:

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

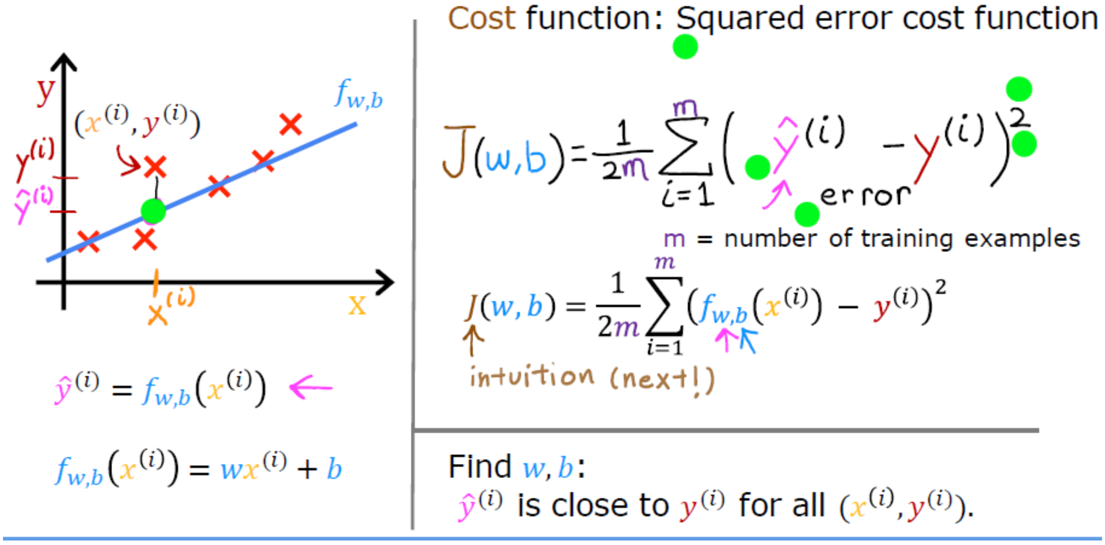

Figure 4: Calculating the cost function or mean square error

Cost function intuition:

> 💡 **Here we set b=0 for better understanding of the model**

- Model: $f_{w,b}(x) = wx + b$, parameters: w, b
- Cost function: $J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(f_{w,b}(x^{(i)}) - y^{(i)})^2$
- Our goal: $\min_{w,b} J(w,b)$
- Understanding how cost function acts: See [page 41-44] (check pdf link embeded at start)

- We simplify our cost function to only be a factor of w (so $f_w(x) = wx$)
- Using training set $\{(1,1),(2,2),(3,3)\}$
- By graphing w vs. $J(w)$, the graph is a parabola

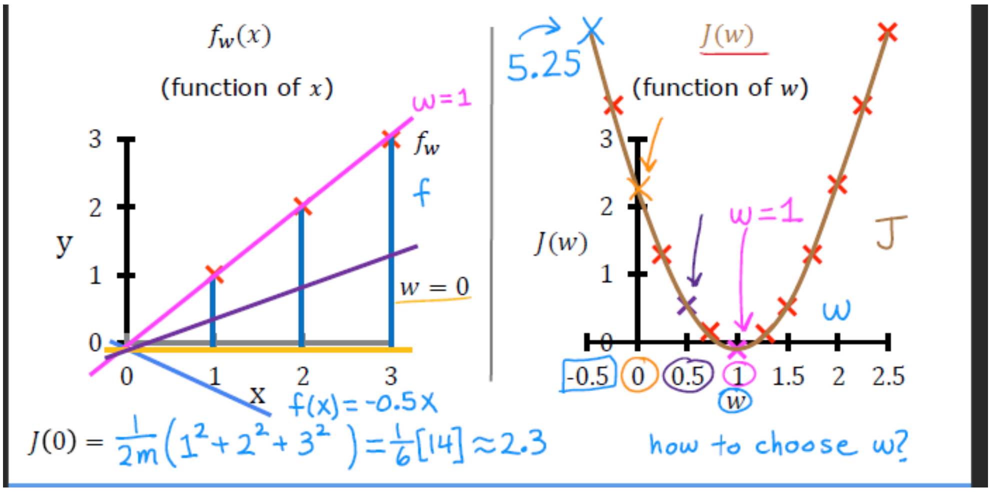

Figure 5: Cost function graph where b=0

Visualizing the cost function:

> 💡 **Now visualizing the model with both parameters w and b**

- Model: $\mathbf{f}_{w,b}(x) = wx + b$, where parameters are : w, b
- Cost Function: $J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(f_{w,b}(x^{(i)}) - y^{(i)})^2$
- Objective: $\min_{w,b} J(w,b)$
- We get the 2D "soup bowl looking" (Figure 6) group of $J(w,b)$ when we plot the graph.

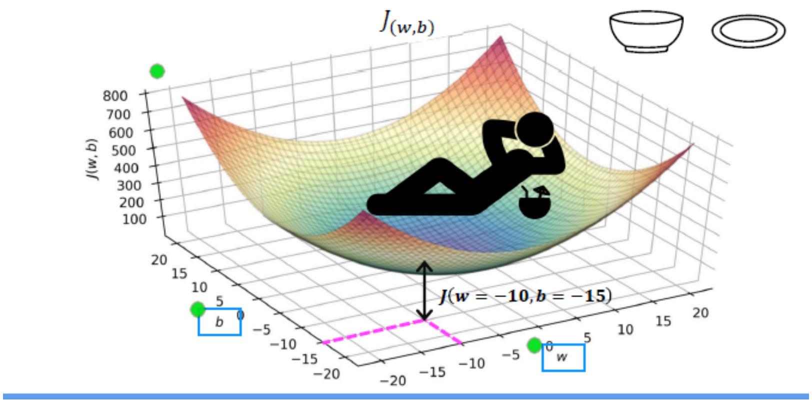

Figure 6: Cost function J_w, b as a function of b,w

- We can use a contour plot to visualize the 2D graph. Each ring in the contour has equivalent cost function values.

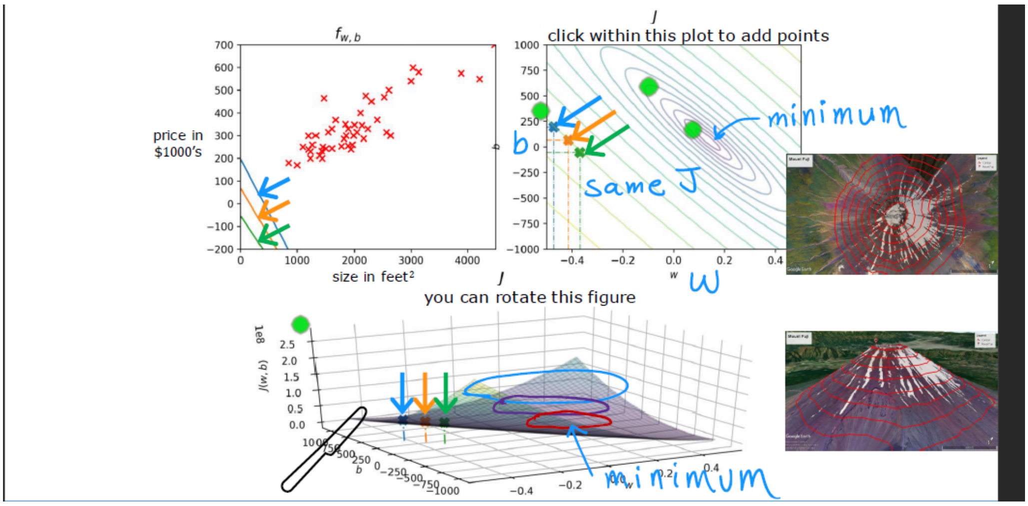

Figure 7: Contour plot of $J_{w,b}$, b, w

- In Figure 7**, we have 3 different functions which give us the same cost function value $J_{(w,b)}$. So, we need to find different values of w and b so that cost function $J_{(w,b)}$ remains constant.**

> 💡 In order for linear regression to work, we will minimize $J_{(w,b)}$. So in other words, we **need to find value of w,b in such a way that $J_{(w,b)}$** minimum.

- To do this, we use ***gradient descent algorithm***. This algorithm is one of the most important algorithms in machine learning. Gradient descent and variations on gradient descent are used to train, not just linear regression, but some of the biggest and most complex models in all of AI.