

## The Business Problem

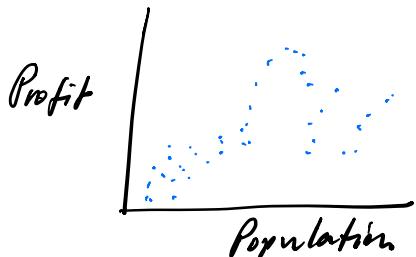
- CEO of a restaurant chain
- Thinking about how to expand the business
- Have data on current restaurants
  - \* population of town in which restaurant is located
  - \* profit from restaurant per quarter
- Where should the next 5 restaurants be opened to maximize profit growth?

## The Data

Population (10,000)	Profit (\$100,000)
6.11	17.59
5.53	9.13
8.52	13.67
.	.
:	:
12.68	15.34

Structure of the data

{ food-truck-profits.txt  
97 rows  
numerical information (not categorical)  
No missing information



Step 1

Scatterplot of the dataset  
to visualize it

## STEP 2

Define the task  
(w.r.t. the dataset )

Input → the population size

Output → the profit

Task → Given a population size,  
predict the profit.

### STEP 3

Define the Model

Feature = input = population size

(later we'll see how features can differ from inputs)

Transform the feature into an output

$(w_0 \times x_0^{(1)}) + (w_1 \times x_1^{(1)}) = \hat{y}^{(1)}$

Parameter 1	Constant feature	Parameter 2	Feature 1	Predicted output	Actual Output
$w_0$	$x_0$	$w_1$	$x_1$	$\hat{y}^{(1)}$	$y^{(1)}$
$w_0$	1	$w_1$	$x_1^{(1)}$	$\hat{y}^{(1)}$	$y^{(1)}$
$w_0$	1	$w_1$	$x_1^{(2)}$	$\hat{y}^{(2)}$	$y^{(2)}$
M rows					
$m = 97$	:	:	:	:	:
$w_0$	1	$w_1$	$x_1^{(97)}$	$\hat{y}^{(97)}$	$y^{(97)}$

These values don't change from row to row

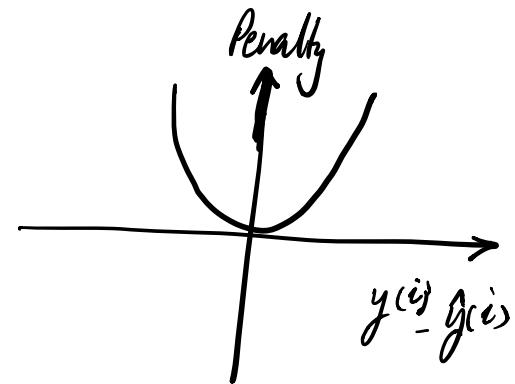
These values change from row to row

#### STEP 4

Define the penalty for getting it wrong.

$$\text{Penalty} = \left( \hat{y}^{(i)} - \hat{y}^{(i)} \right)^2$$

for each row

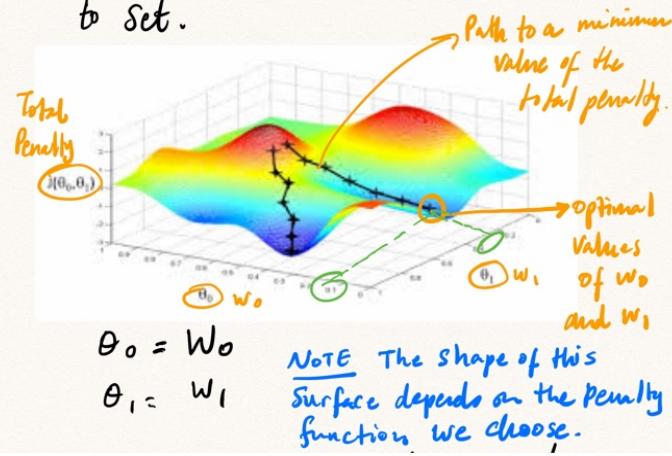


The more we get it wrong, the more we're penalized.

Total Penalty for the entire data set = Sum of penalty of each row.

| **QUESTION** | What values of  $w_0$  and  $w_1$ , will minimize the total penalty ?

Imagine we only had  $w_0$  and  $w_1$  to set.



Use gradient descent to solve the numerical optimization problem.

- start at an arbitrary point on the surface - pick arbitrary values for  $w_0$  and  $w_1$
- From this point, take a step in the direction where the slope is descending most steeply.
- Do this again and again until you reach a minimum (the "valley floor")

## Recap

Model - transforms the input(s) into the output.

Parameters - these are the fixed values (e.g.,  $w_0, w_i$ ) of the model.

Penalty - for each row of the dataset, what's the penalty/cost/price for being wrong?

## Recap (Contd.)

Total Penalty - Sum of the penalty of each row of the dataset.

Problem - Find the parameter

values of the model ( $w_0$  and  $w_i$  values) that minimize the total penalty.

Approach to Solving the Problem - Gradient Descent.

## Hyper-Parameters

Gradient descent can produce different results based on how you set the values of the following parameters.

- \* Learning Rate. The size of the jump taken at each step.
- \* The total number of jumps taken.

Every machine learning problem  
has hyperparameters.

Why? Because every machine  
learning problem is a giant optimization  
problem - find the optimal values  
of the model parameters that  
minimizes the total penalty.

To find these optimal values of  
 $w_0$  and  $w_i$  (for example), we  
use a learning algorithm like  
gradient descent.

Finding the right hyperparameters  
is a separate quest - one  
that must be solved every time  
you apply machine learning to  
solve a problem.

In a few sessions we'll see  
how this problem is tackled.