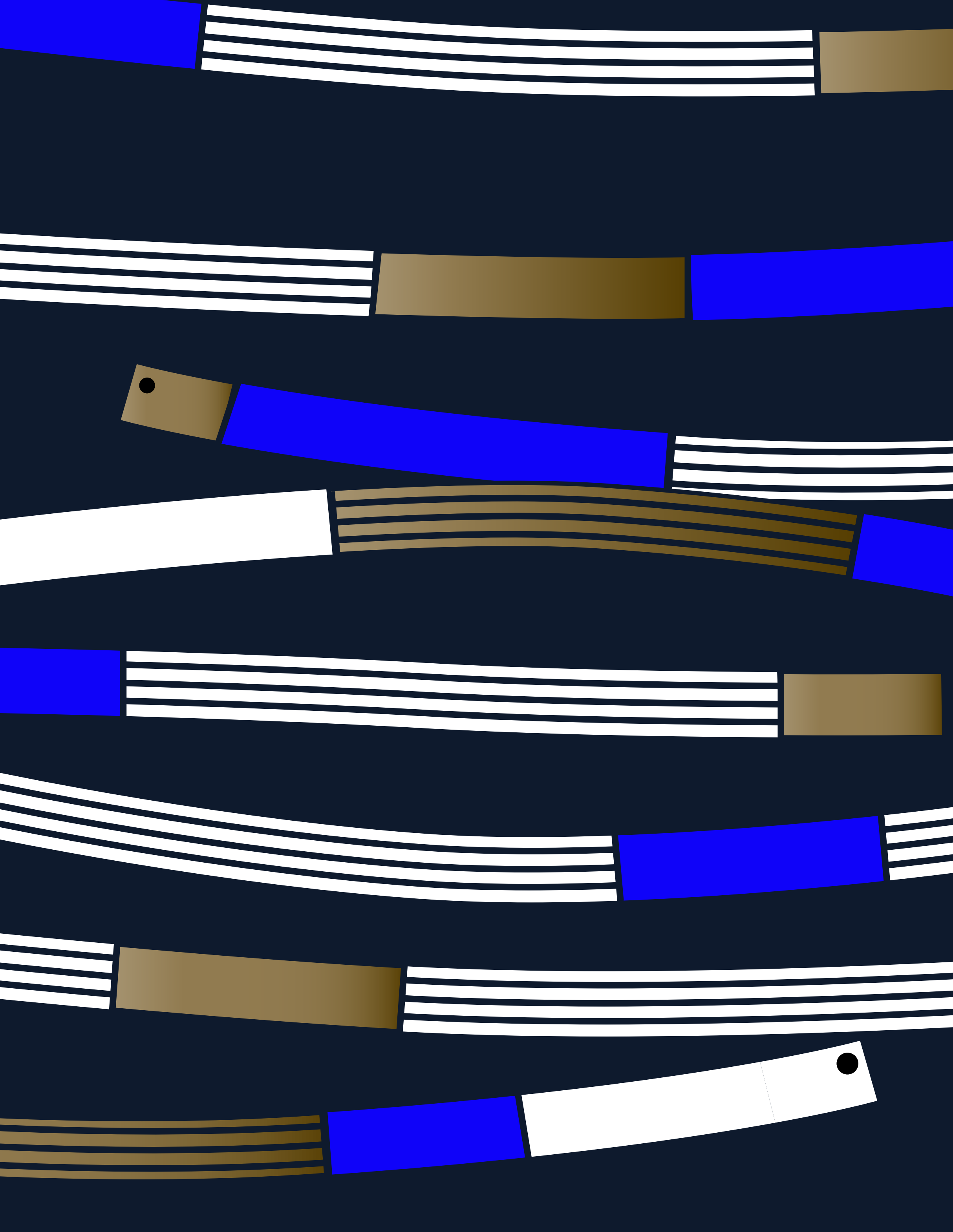


3.1 线性回归模型

Data table

size in feet²	price in \$1000's
2104	400
1416	232
1534	315
852	178
...	...
3210	870

监督学习 — 回归模型 — 预测数字
eg.
区分分类模型 (识别) 猫/狗句: 输出少量离散数字结果)

训练集

Terminology

Training set: Data used to train the model

	x size in feet²	y price in \$1000's
(1)	2104	400
(2)	1416	232
(3)	1534	315
(4)	852	178
...	...	...
(47)	3210	870

$m = 47$

Notation:
x = "input" variable
feature
y = "output" variable
"target" variable
m = number of training examples

(x, y) = single training example

$x^{(1)} = 2104$ $y^{(1)} = 400$

$(x^{(1)}, y^{(1)}) = (2104, 400)$

$(x^{(i)}, y^{(i)})$ 第i个训练样本

$(x^{(i)}, y^{(i)})$ = i^{th} training example

index (1st, 2nd, 3rd ...)

$x^{(2)} = 1416$ $x^{(2)} \neq x^2$ not exponent

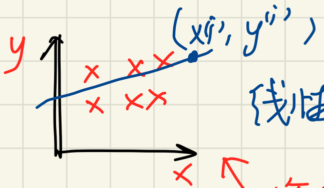
$x \rightarrow \boxed{f} \rightarrow \hat{y}$ 对y的估计

输入 模型 输出

target 真实值

How to represent f?

$$f_{(x)} = f_{w,b}(x) = wx + b$$



线性回归

特征量

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

cost function

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

m 训练样本数量

误差

平方误差代价函数

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

J of wb　　cost j　　?

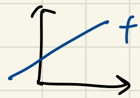
较大　　　较小

3.4 理解代价函数

Modle:

$$f_{w,b}(x) = wx + b$$

参数: w、b



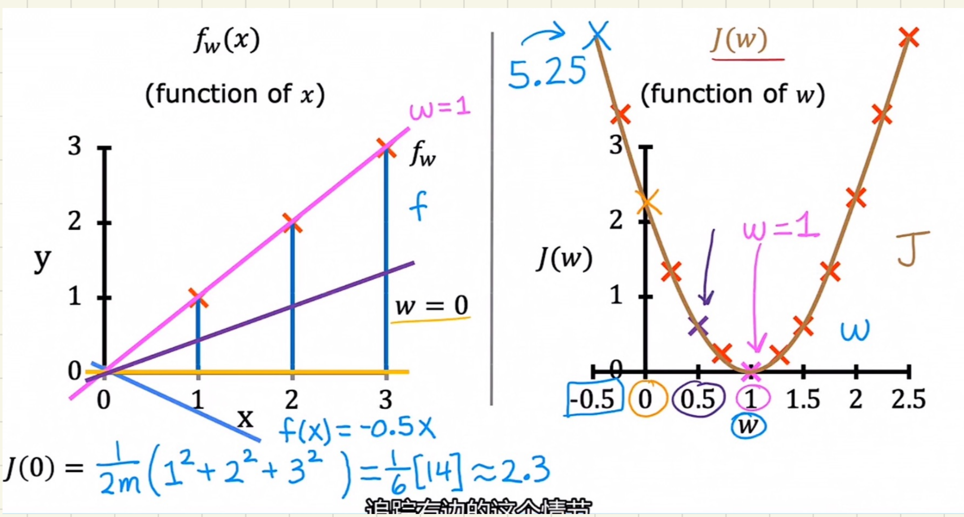
cost function:

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

goal:

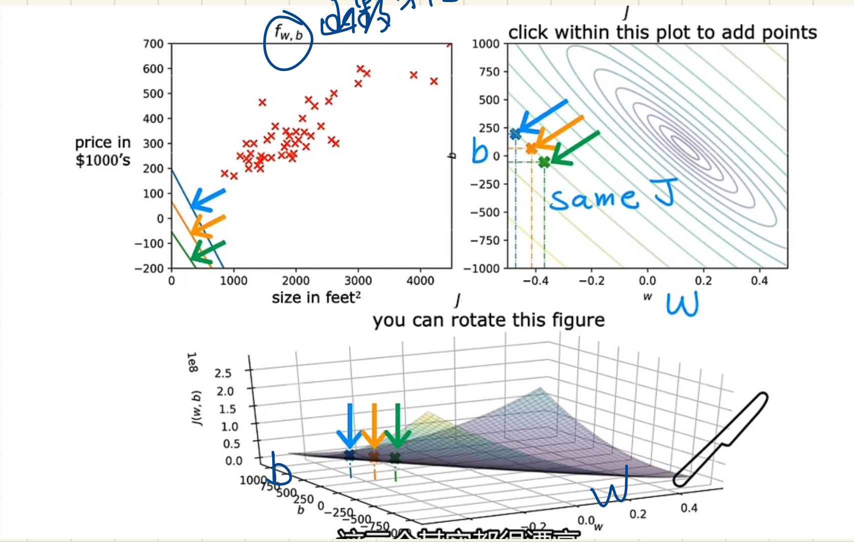
minimize $J(w,b)$

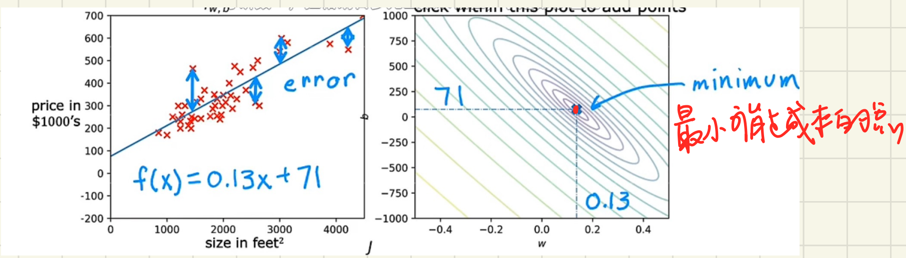
最小化



3.4 可视化举例

函数 $wx+b$





4.1 梯度下降 gradient descent algorithm

步长 learning rate $\alpha \in (0,1)$

update:

$$\underset{\text{old}}{w} = w - \alpha \boxed{\frac{\partial J(w,b)}{\partial w}}$$ → derivative 方向

$$b = b - \alpha \frac{\partial J(w,b)}{\partial b}$$

Repeat this two update steps util convergence
收敛
达到局部最小值

Correct: Simultaneous update

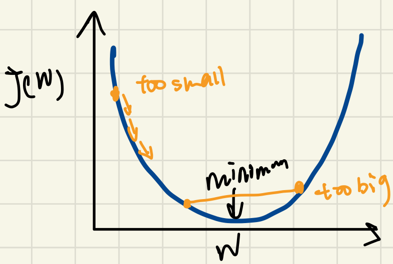
保证同时更新

$$tmp\_w = w - \alpha \frac{\partial J(w,b)}{\partial w}$$

$$tmp\_b = b - \alpha \frac{\partial J(w,b)}{\partial b}$$

$$w = tmp\_w$$

$$b = tmp\_b$$



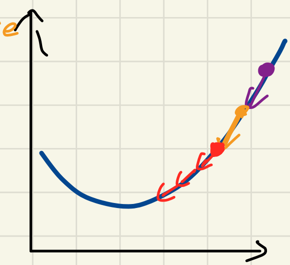
α is too Small
Gradient descent Slow

α large
: overshoot
- never reach minimum
- fail to converge

$$w := w - \alpha \boxed{\frac{dJ(w)}{dw}} = 0$$

Can reach local minimum with fixed learning rate

$$w := w - \overbrace{\overbrace{\boxed{\alpha \frac{dJ(w)}{dw}}}^{\text{not as large}}}^{\text{smaller}} \text{ large}$$



Near a local minimum

- Derivative ↓
- Update steps ↓

4.5. Gradient Descent for Linear Regression

Linear regression model

$$f_{w,b}(x) = wx + b$$

Cost function

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

Gradient descent algorithm

repeat until convergence {

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

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

}