به نام یگانه معبود بخشنده مهربان

مبانی یادگیری ماشین

Machine Learning Foundations

گروه هوش مصنوعی، دانشکده مهندسی کامپیوتر، دانشگاه اصفهان

ترم اول سال تحصیلی ۰۲ - ۰۳

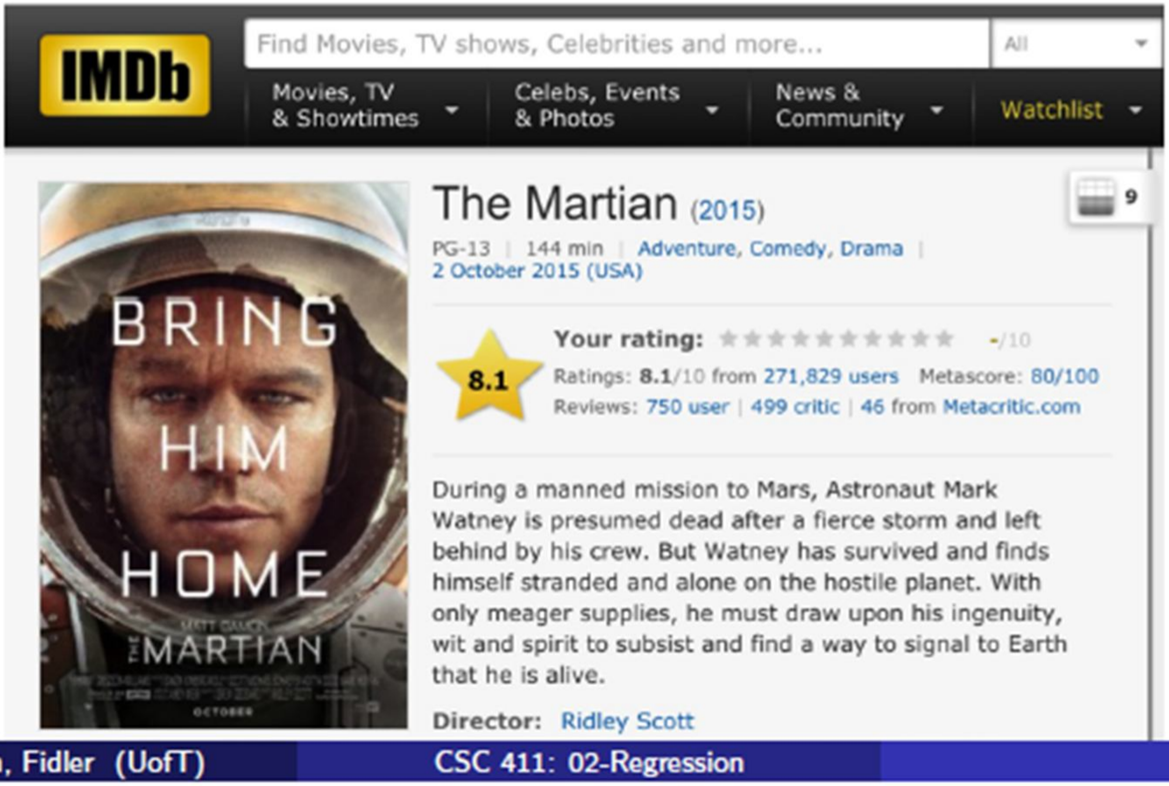
ارائه دهنده : پیمان ادیبی

رگرسیون خطی

Linear Regression

رگرسیون: مثال

- What should I watch this Friday?
- **Goal**: Predict movie rating automatically!
- **Goal:** How many followers will I get?
- **Goal:** Predict the price of the house



The Martian (2015)

PG-13 | 144 min | Adventure, Comedy, Drama |
2 October 2015 (USA)

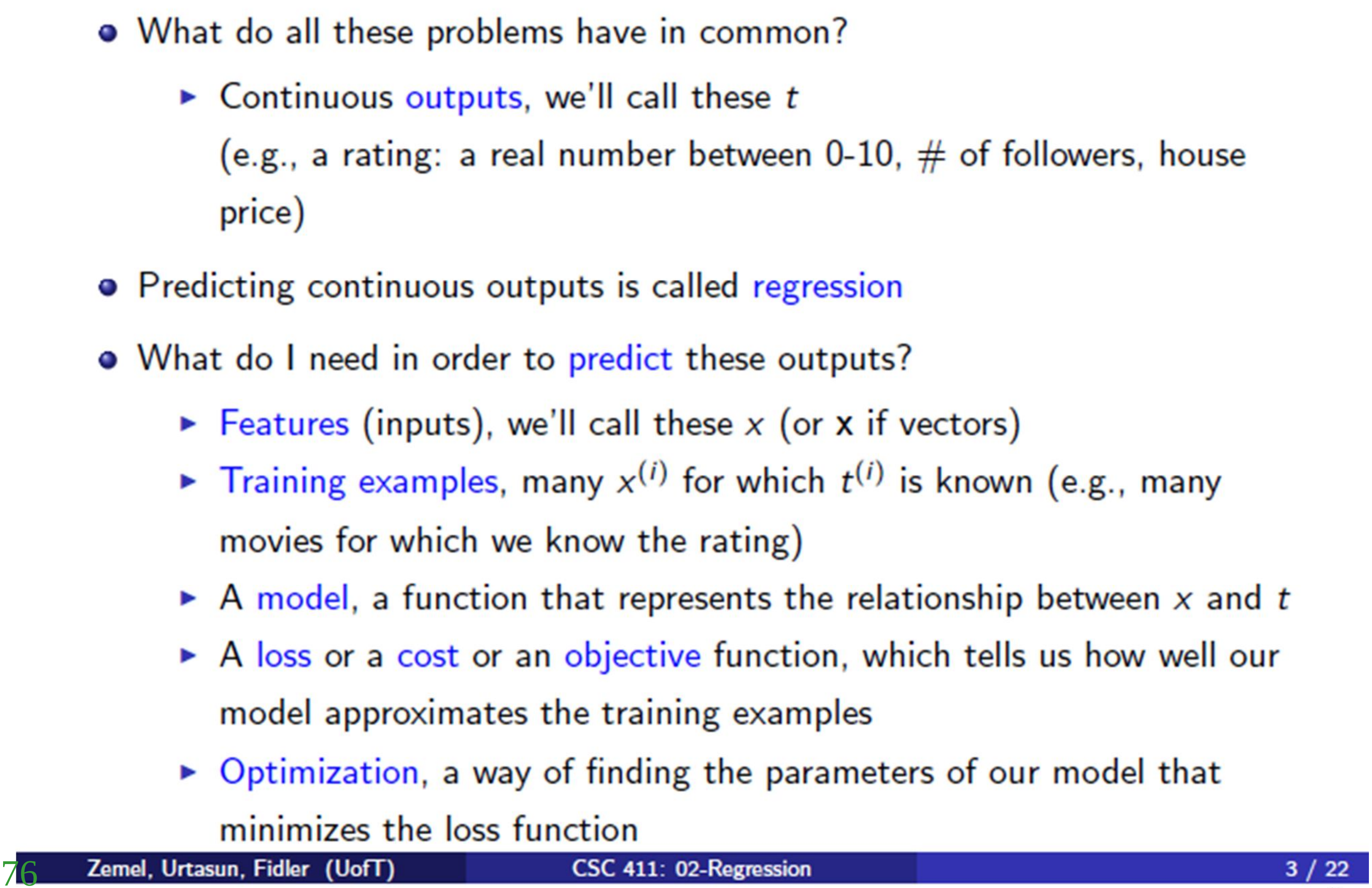
8.1 **Your rating:** ★★★★★★★★★★ -/10

Ratings: **8.1**/10 from 271,829 users Metascore: 80/100
Reviews: 750 user | 499 critic | 46 from Metacritic.com

During a manned mission to Mars, Astronaut Mark Watney is presumed dead after a fierce storm and left behind by his crew. But Watney has survived and finds himself stranded and alone on the hostile planet. With only meager supplies, he must draw upon his ingenuity, wit and spirit to subsist and find a way to signal to Earth that he is alive.

Director: Ridley Scott

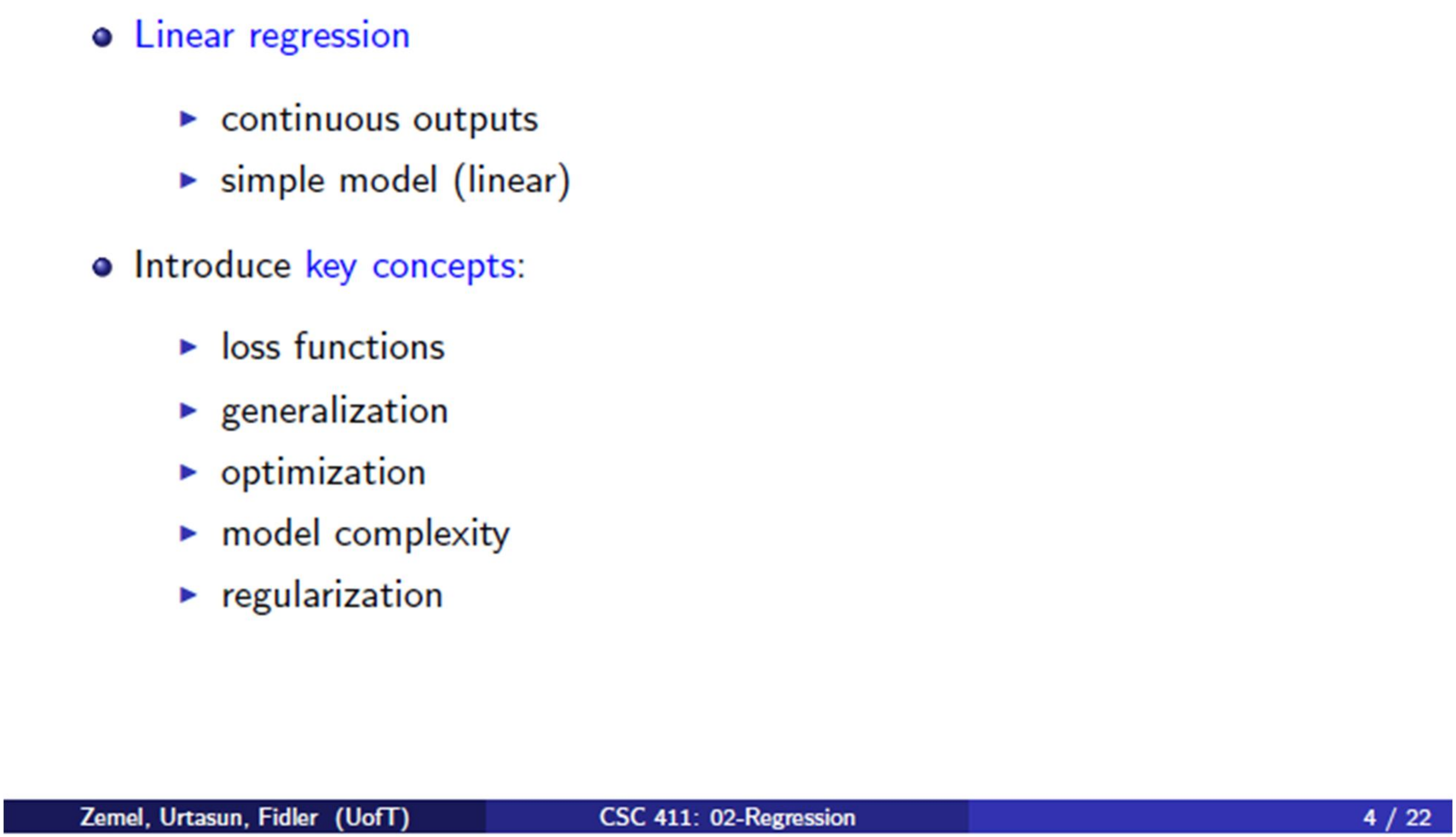
رگرسیون: معرفی مسأله و راه حل



- What do all these problems have in common?
  - Continuous outputs, we'll call these t
    (e.g., a rating: a real number between 0-10, # of followers, house price)
- Predicting continuous outputs is called regression
- What do I need in order to predict these outputs?
  - Features (inputs), we'll call these x (or $\mathbf{x}$ if vectors)
  - Training examples, many $x^{(i)}$ for which $t^{(i)}$ is known (e.g., many movies for which we know the rating)
  - A model, a function that represents the relationship between x and t
  - A loss or a cost or an objective function, which tells us how well our model approximates the training examples
  - Optimization, a way of finding the parameters of our model that minimizes the loss function

Zemel, Urtasun, Fidler (UofT) CSC 411: 02-Regression 3 / 22

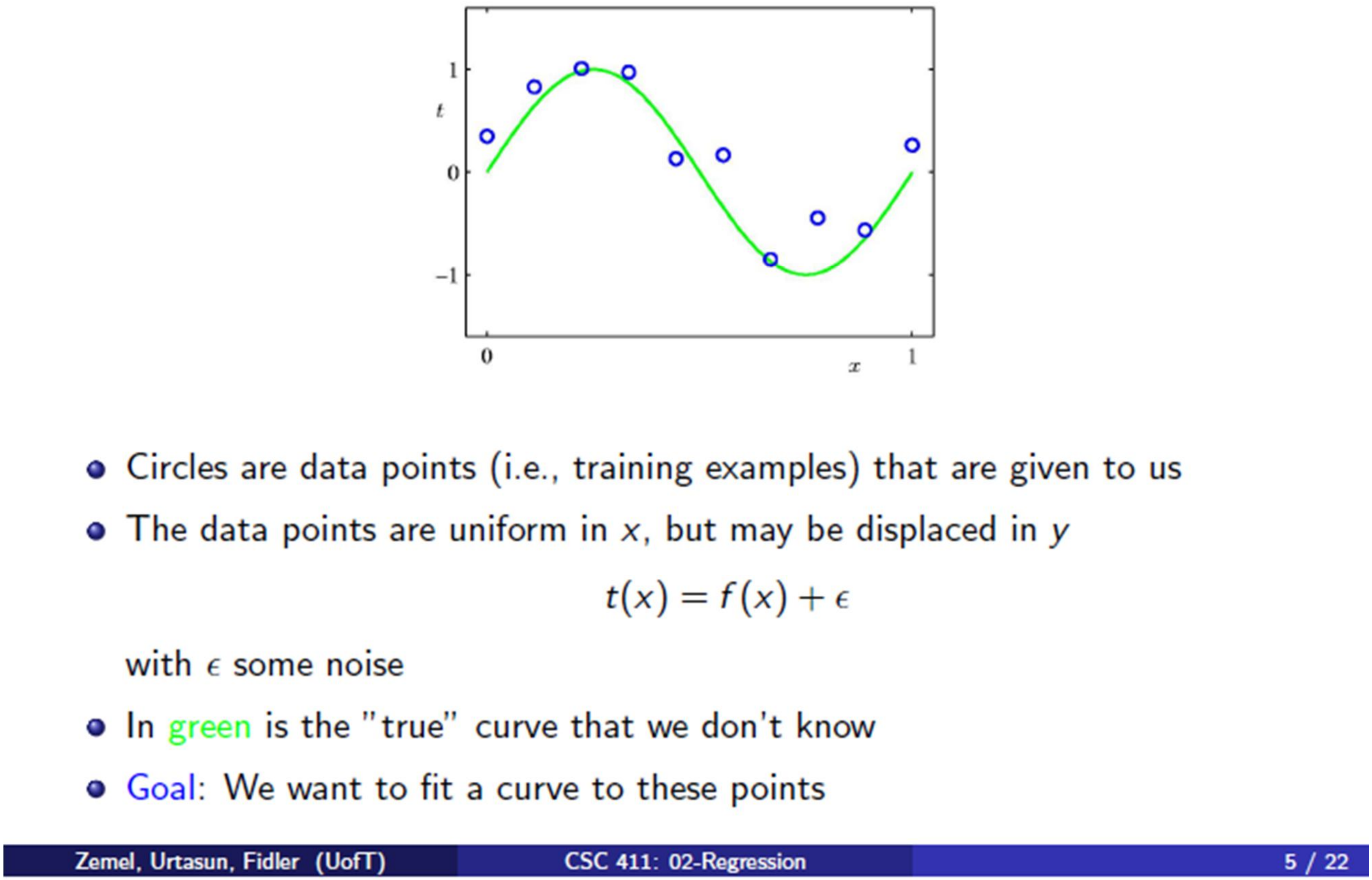
رگرسیون خطی



- Linear regression
  - continuous outputs
  - simple model (linear)
- Introduce key concepts:
  - loss functions
  - generalization
  - optimization
  - model complexity
  - regularization

Zemel, Urtasun, Fidler (UofT) CSC 411: 02-Regression 4 / 22

رگرسیون خطی: مثال ساده یک بعدی



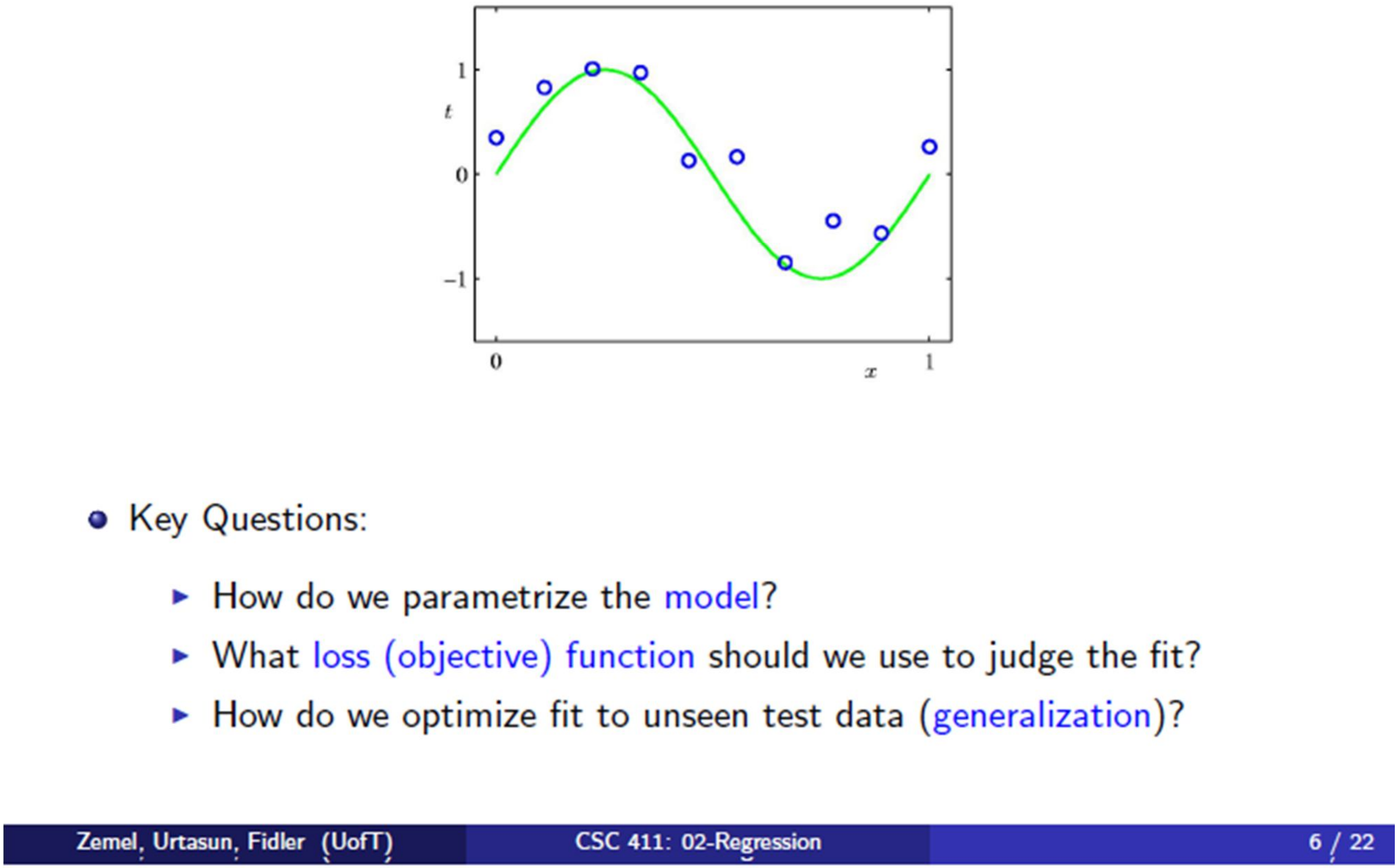
- Circles are data points (i.e., training examples) that are given to us
- The data points are uniform in x, but may be displaced in y

$$t(x) = f(x) + \epsilon$$

with ϵ some noise

- In green is the "true" curve that we don't know
- Goal: We want to fit a curve to these points

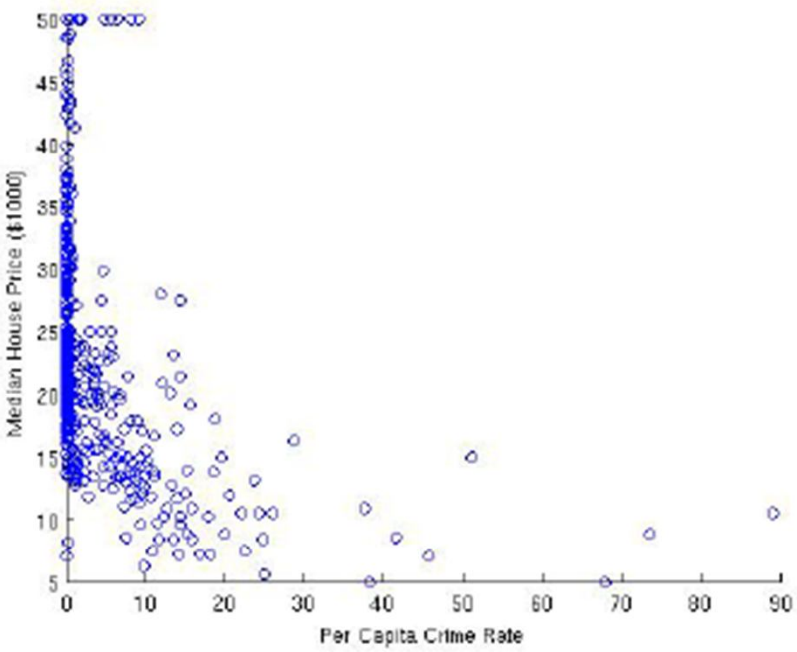
رگرسیون خطی: مثال ساده یک بعدی



- Key Questions:
  - How do we parametrize the model?
  - What loss (objective) function should we use to judge the fit?
  - How do we optimize fit to unseen test data (generalization)?

رگرسیون خطی: مثال ساده یک بعدی

- Estimate median house price in a neighborhood based on neighborhood statistics
- Look at first possible attribute (feature): per capita crime rate



- Use this to predict house prices in other neighborhoods
- Is this a good input (attribute) to predict house prices?

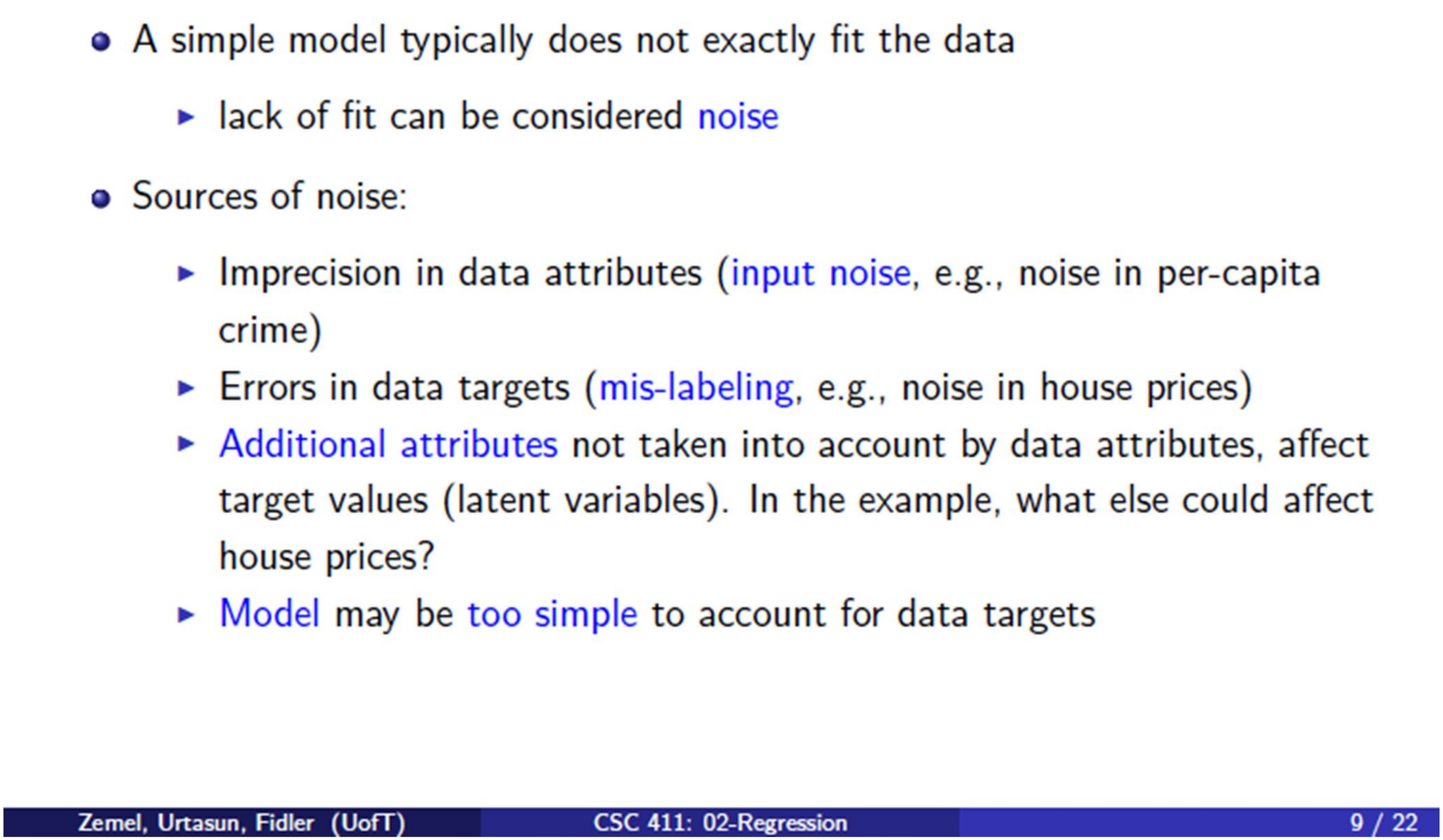
بازنمایی داده

- Data is described as pairs $\mathcal{D} = \{(x^{(1)}, t^{(1)}), \cdots, (x^{(N)}, t^{(N)})\}$
  - $x \in \mathbb{R}$ is the input feature (per capita crime rate)
  - $t \in \mathbb{R}$ is the target output (median house price)
  - $^{(i)}$ simply indicates the training examples (we have N in this case)
- Here t is continuous, so this is a regression problem
- Model outputs y, an estimate of t

$$y(x) = w_0 + w_1 x$$

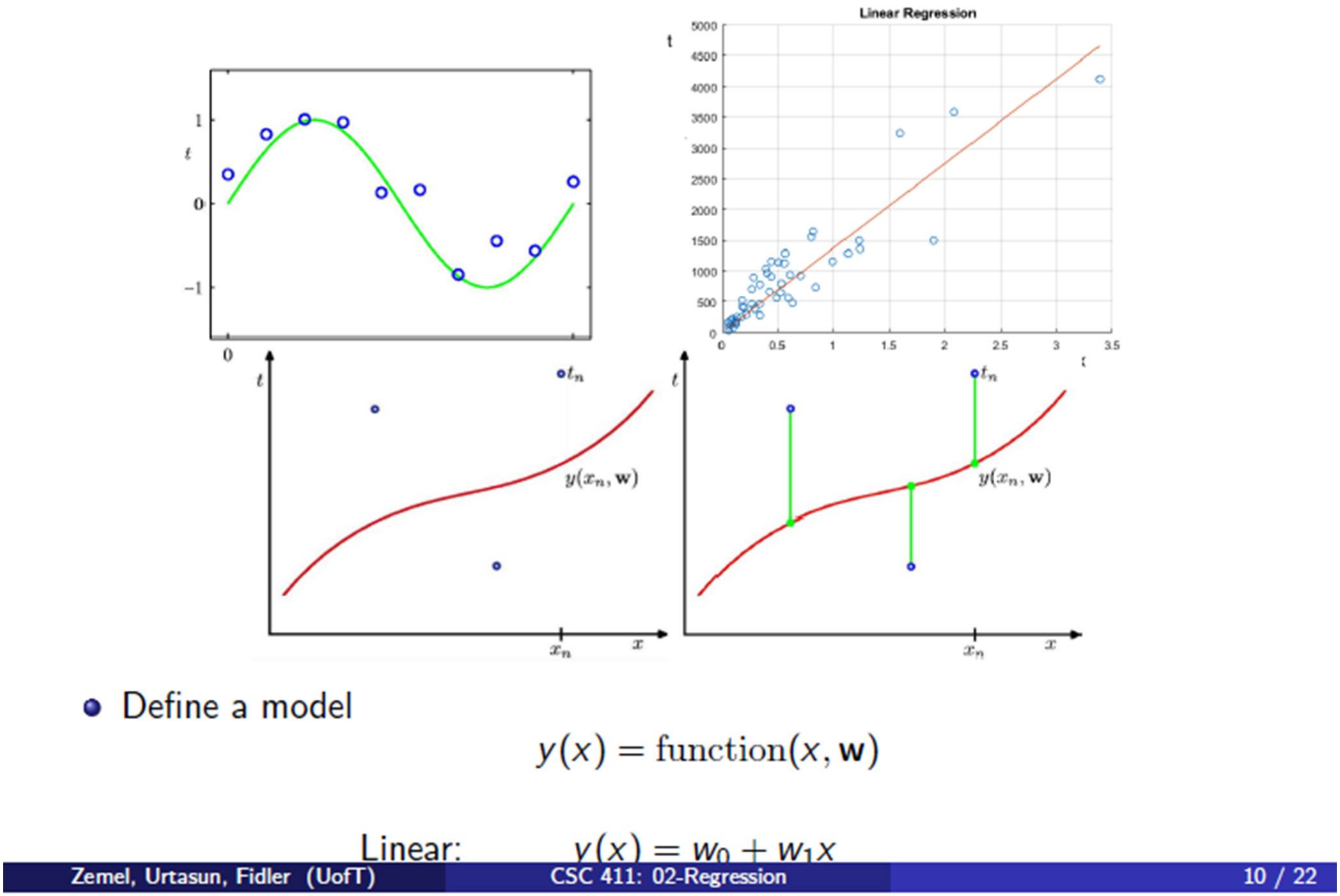
- What type of model did we choose?
- Divide the dataset into training and testing examples
  - Use the training examples to construct hypothesis, or function approximator, that maps x to predicted y
  - Evaluate hypothesis on test set

بازنمایی داده



- A simple model typically does not exactly fit the data
  - lack of fit can be considered noise
- Sources of noise:
  - Imprecision in data attributes (input noise, e.g., noise in per-capita crime)
  - Errors in data targets (mis-labeling, e.g., noise in house prices)
  - Additional attributes not taken into account by data attributes, affect target values (latent variables). In the example, what else could affect house prices?
  - Model may be too simple to account for data targets

رگرسیون کمترین مربعات



- Define a model

$$y(x) = \text{function}(x, \mathbf{w})$$

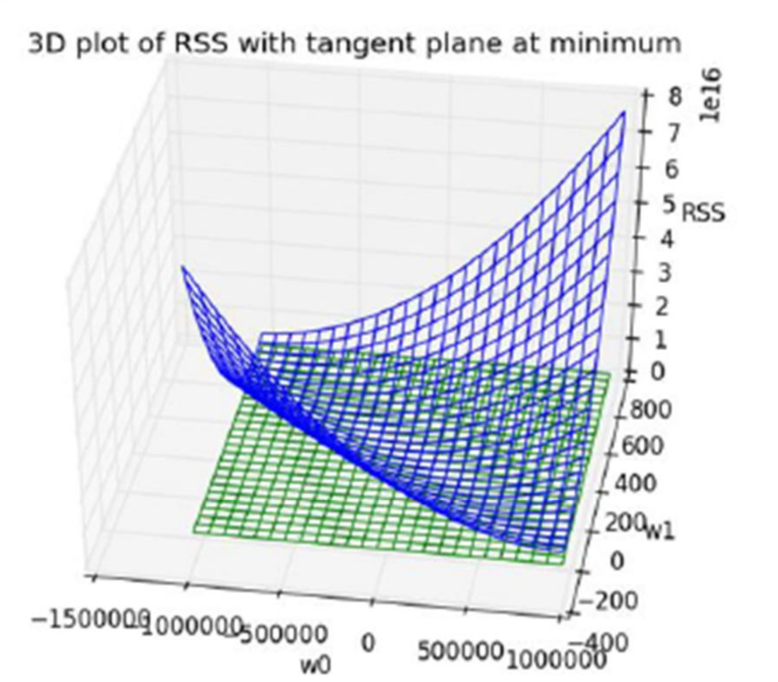
Linear: $y(x) = w_0 + w_1 x$

Zemel, Urtasun, Fidler (UofT) CSC 411: 02-Regression 10 / 22

بهینه سازی تابع هدف

Residual sum of squares (RSS)

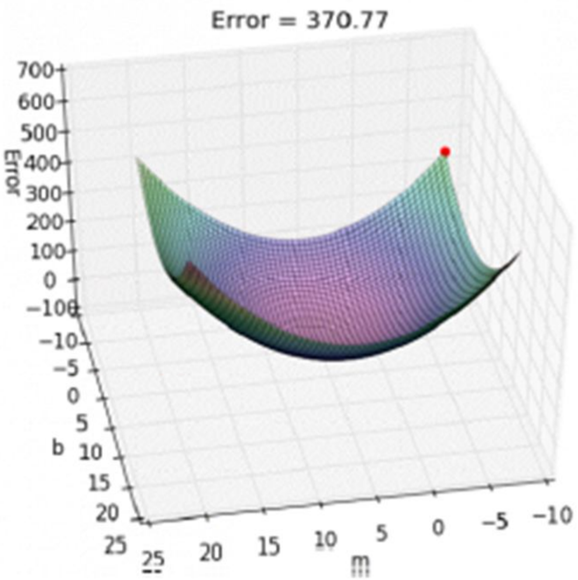
RSS is a function with inputs w_0, w_1, different settings have different RSS for a dataset

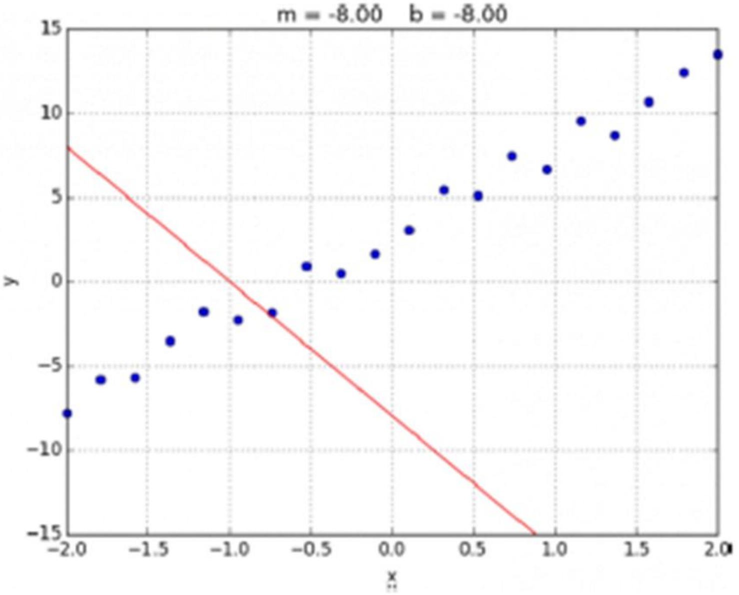


$$
\begin{aligned}
\hat{w}_0, \hat{w}_1 &= \min_{w_0,w_1} RSS(w_0, w_1) \\
&= \min_{w_0,w_1} \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i)\right)^2
\end{aligned}
$$

Unfortunately, we can’t try it out on all possible settings ☹

بهینه سازی تابع هدف

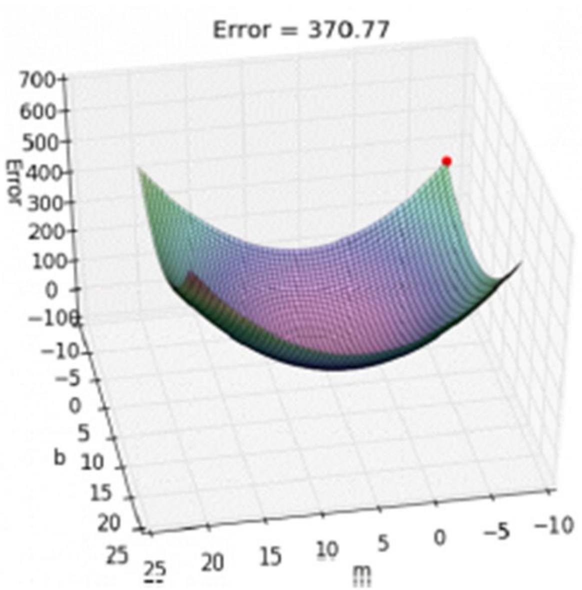


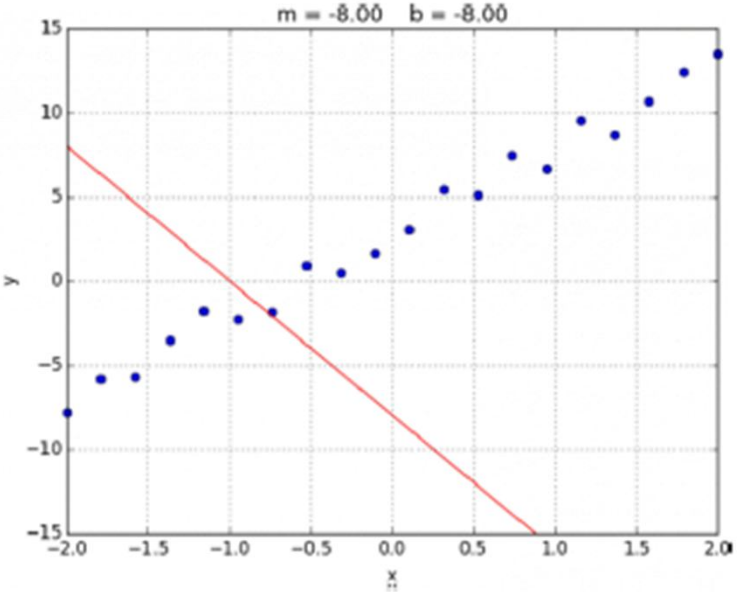


Instead of computing all possible points to find the minimum, just start at one point and "roll" down the hill.
Use the gradient (slope) to determine which direction is down.

```
start at some (random) point $w^{(0)}$ when $t = 0$
while we haven't converged:
        $w^{(t+1)} = w^{(t)} - \eta \nabla RSS(w^{(t)})$
```

بهینه سازی تابع هدف

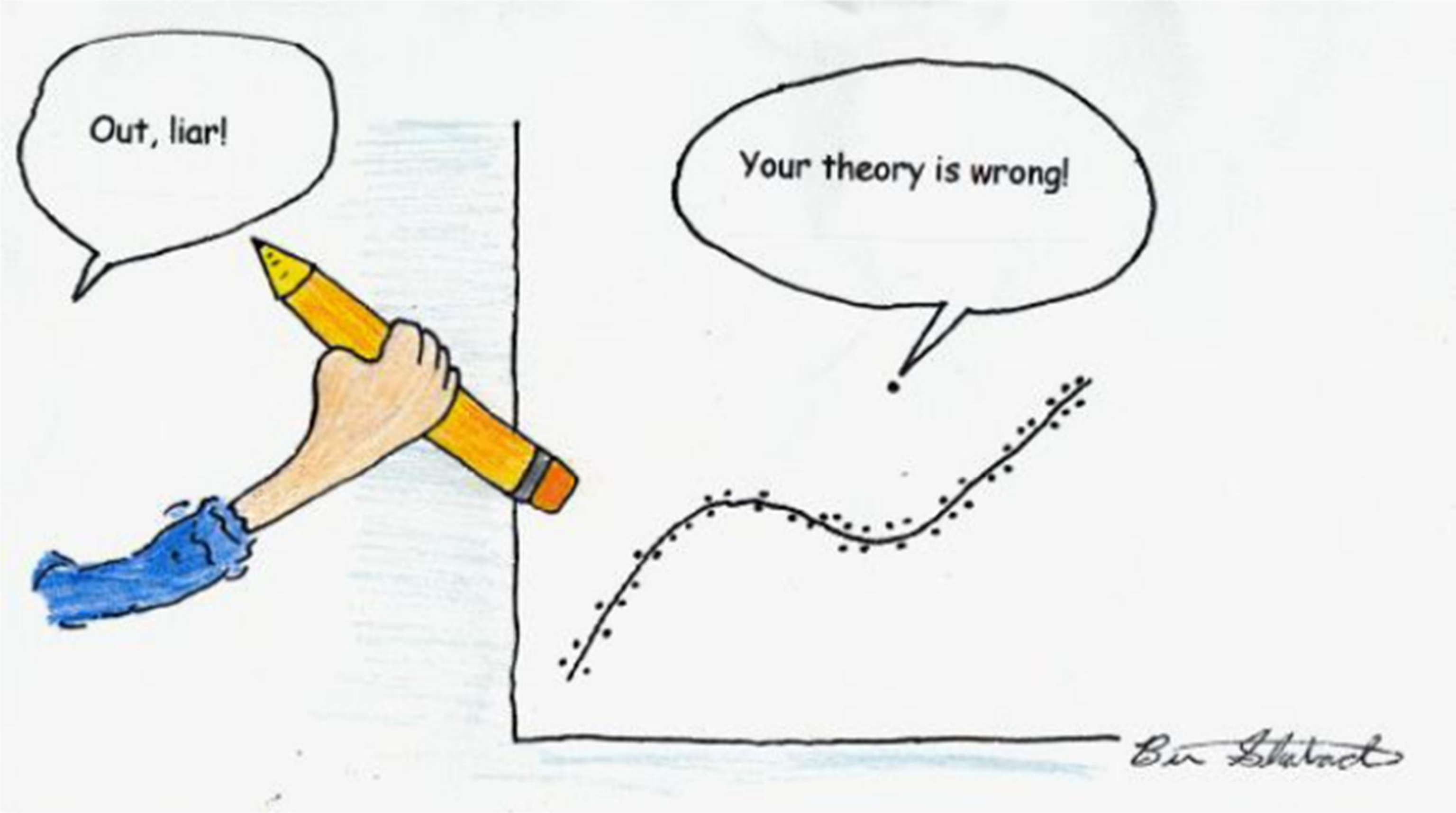




Instead of computing all possible points to find the minimum,
just start at one point and “roll” down the hill.
Use the gradient (slope) to determine which direction is down.

```
start at some (random) point $w^{(0)}$ when $t = 0$
while we haven't converged:
        $w^{(t+1)} = w^{(t)} - \eta \nabla RSS(w^{(t)})$
```

بهینه سازی تابع هدف

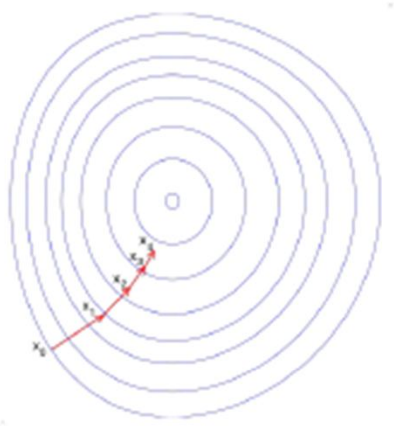


Disclaimer: This is for your comedic entertainment. Please don't actually erase outliers ☺

بهینه سازی تابع هدف

- One straightforward method: gradient descent
  - initialize $\mathbf{w}$ (e.g., randomly)
  - repeatedly update $\mathbf{w}$ based on the gradient

$$\mathbf{w} \leftarrow \mathbf{w} - \lambda \frac{\partial \ell}{\partial \mathbf{w}}$$



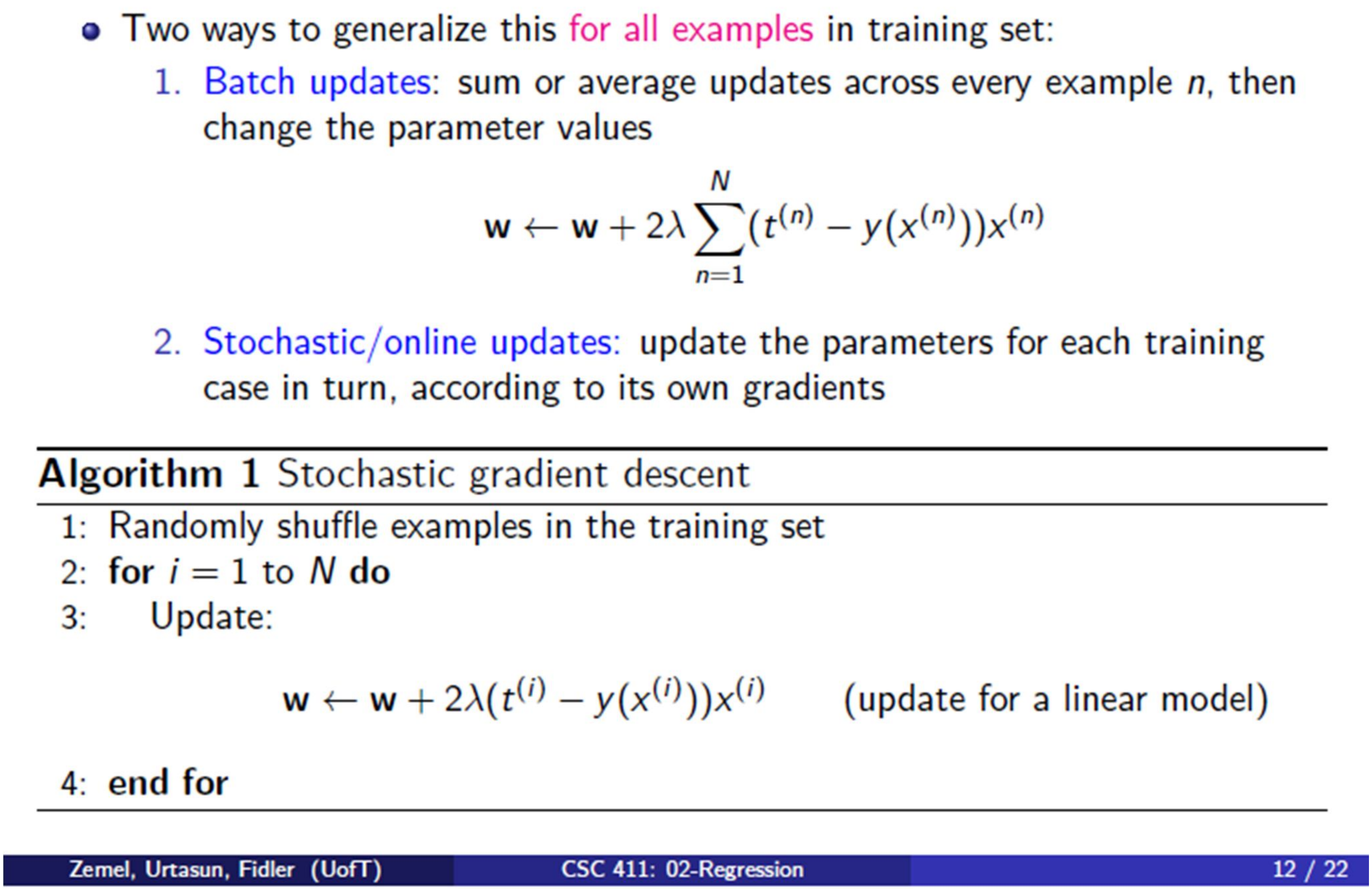
- λ is the learning rate
- For a single training case, this gives the LMS update rule (Least Mean Squares):

$$\mathbf{w} \leftarrow \mathbf{w} + 2\lambda(t^{(n)} - y(x^{(n)}))x^{(n)}$$

$$\mathbf{w} \leftarrow \mathbf{w} + 2\lambda \underbrace{(t^{(n)} - y(x^{(n)}))}_{\text{error}} x^{(n)}$$

- Note: As error approaches zero, so does the update ($\mathbf{w}$ stops changing)

بهینه سازی روی مجموعه آموزشی



- Two ways to generalize this for all examples in training set:
  1. Batch updates: sum or average updates across every example n, then change the parameter values

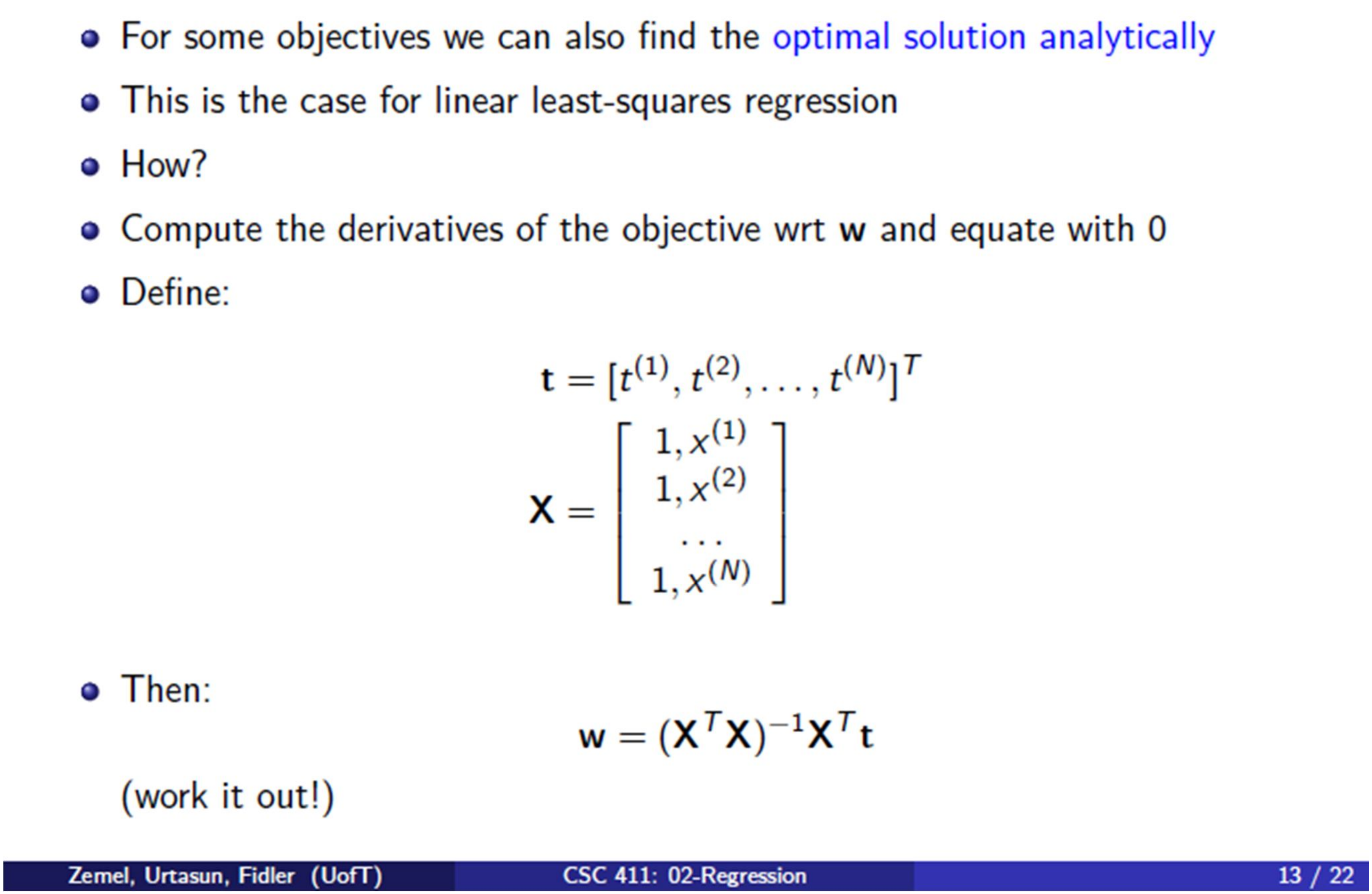
$$\mathbf{w} \leftarrow \mathbf{w} + 2\lambda \sum_{n=1}^{N} (t^{(n)} - y(x^{(n)}))x^{(n)}$$

  2. Stochastic/online updates: update the parameters for each training case in turn, according to its own gradients

Algorithm 1 Stochastic gradient descent

```
Randomly shuffle examples in the training set
for $i = 1$ to $N$ do
    Update:
        $\mathbf{w} \leftarrow \mathbf{w} + 2\lambda(t^{(i)} - y(x^{(i)}))x^{(i)}$    (update for a linear model)
end for
```

وجود پاسخ تحلیلی



- For some objectives we can also find the optimal solution analytically
- This is the case for linear least-squares regression
- How?
- Compute the derivatives of the objective wrt $\mathbf{w}$ and equate with 0
- Define:

$$\mathbf{t} = [t^{(1)}, t^{(2)}, \ldots, t^{(N)}]^T$$

$$\mathbf{X} = \begin{bmatrix} 1, x^{(1)} \\ 1, x^{(2)} \\ \ldots \\ 1, x^{(N)} \end{bmatrix}$$

- Then:

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t}$$

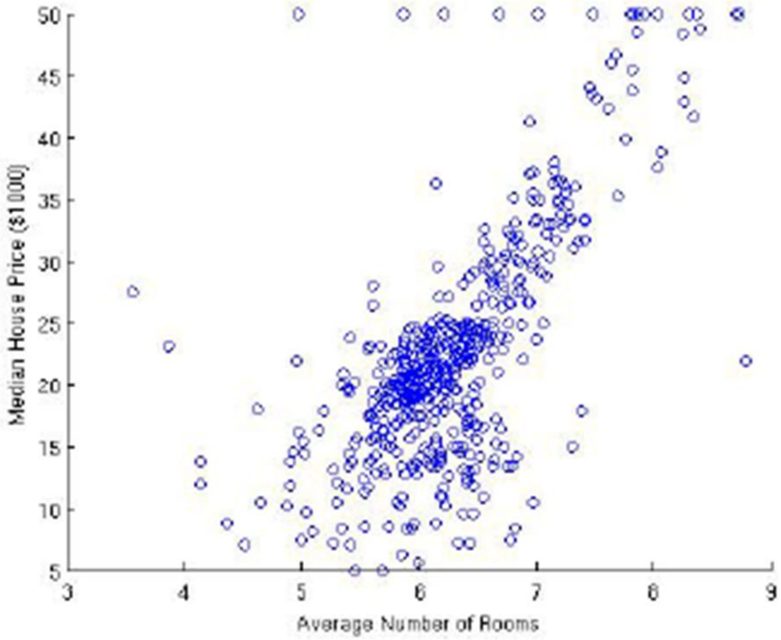
(work it out!)

ورودی های چند بعدی

- One method of extending the model is to consider other input dimensions

$$y(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2$$

- In the Boston housing example, we can look at the number of rooms



رگرسیون خطی با ورودی های چند بعدی

- Imagine now we want to predict the median house price from these multi-dimensional observations
- Each house is a data point n, with observations indexed by j:

$$\mathbf{x}^{(n)} = \left(x_1^{(n)}, \cdots, x_j^{(n)}, \cdots, x_d^{(n)}\right)$$

- We can incorporate the bias w_0 into $\mathbf{w}$, by using $x_0 = 1$, then

$$y(\mathbf{x}) = w_0 + \sum_{j=1}^{d} w_j x_j = \mathbf{w}^T\mathbf{x}$$

- We can then solve for $\mathbf{w} = (w_0, w_1, \cdots, w_d)$. How?
- We can use gradient descent to solve for each coefficient, or compute $\mathbf{w}$ analytically (how does the solution change?)

مدلهای پیچیده تر : برازش با چند جمله ای

- What if our linear model is not good? How can we create a more complicated model?
- We can create a more complicated model by defining input variables that are combinations of components of $\mathbf{x}$
- Example: an M-th order polynomial function of one dimensional feature x:

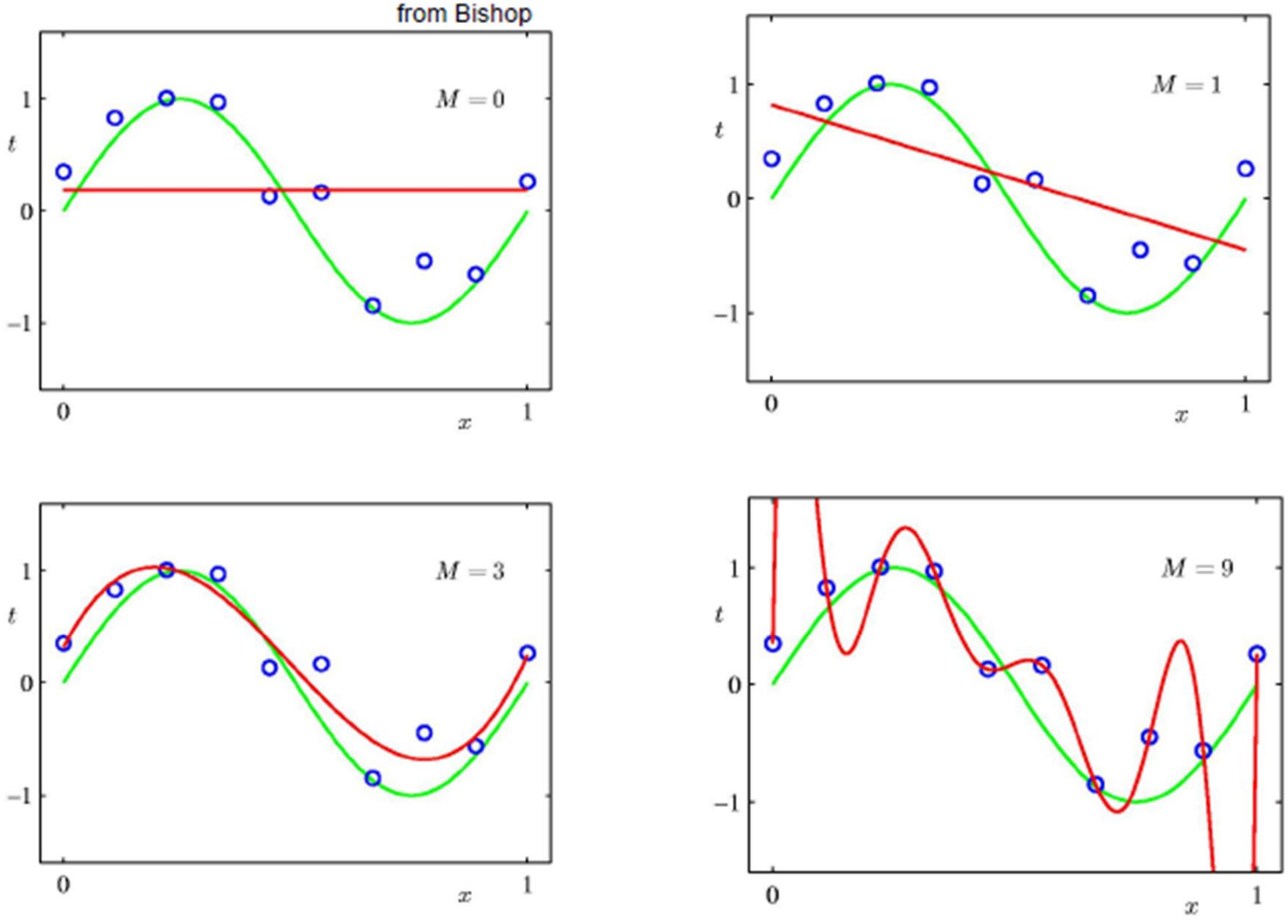
$$y(x, \mathbf{w}) = w_0 + \sum_{j=1}^{M} w_j x^j$$

where x^j is the j-th power of x

- We can use the same approach to optimize for the weights $\mathbf{w}$
- How do we do that?

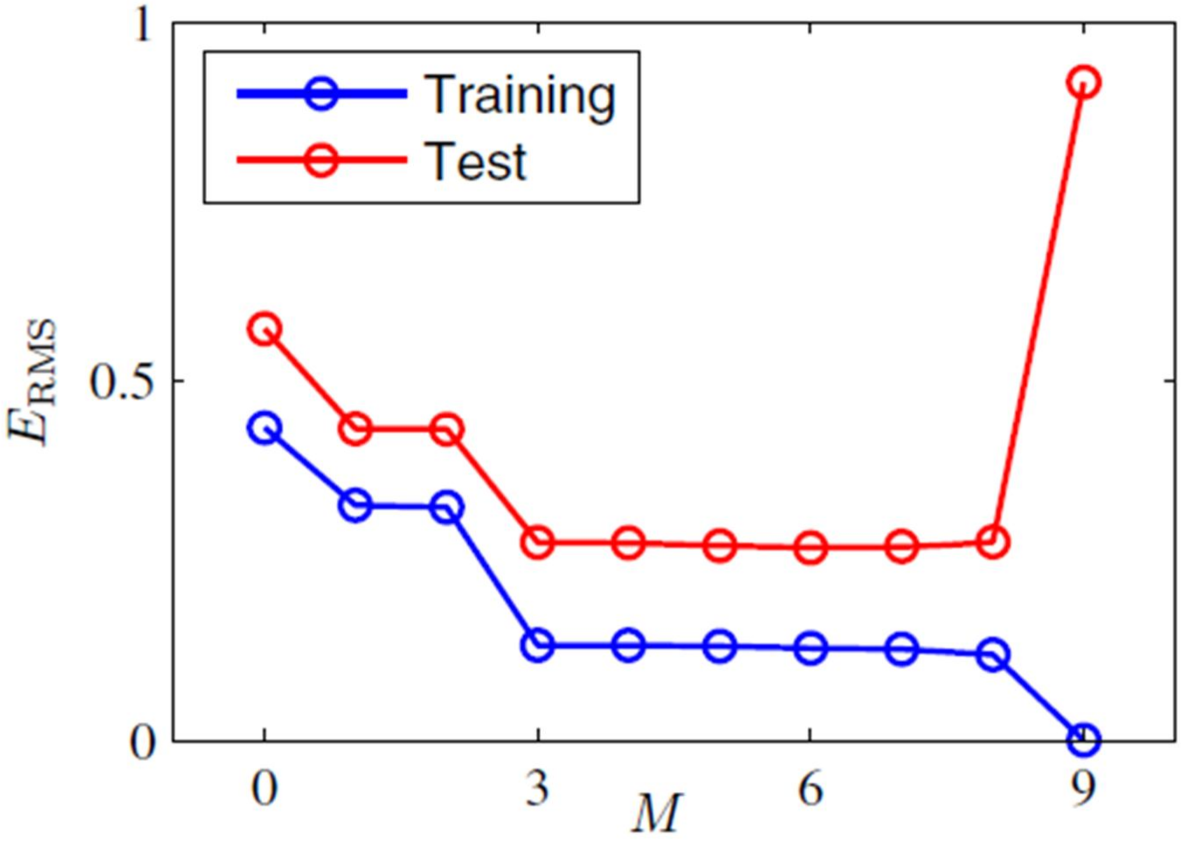
کدام برازش بهتر است؟

from Bishop

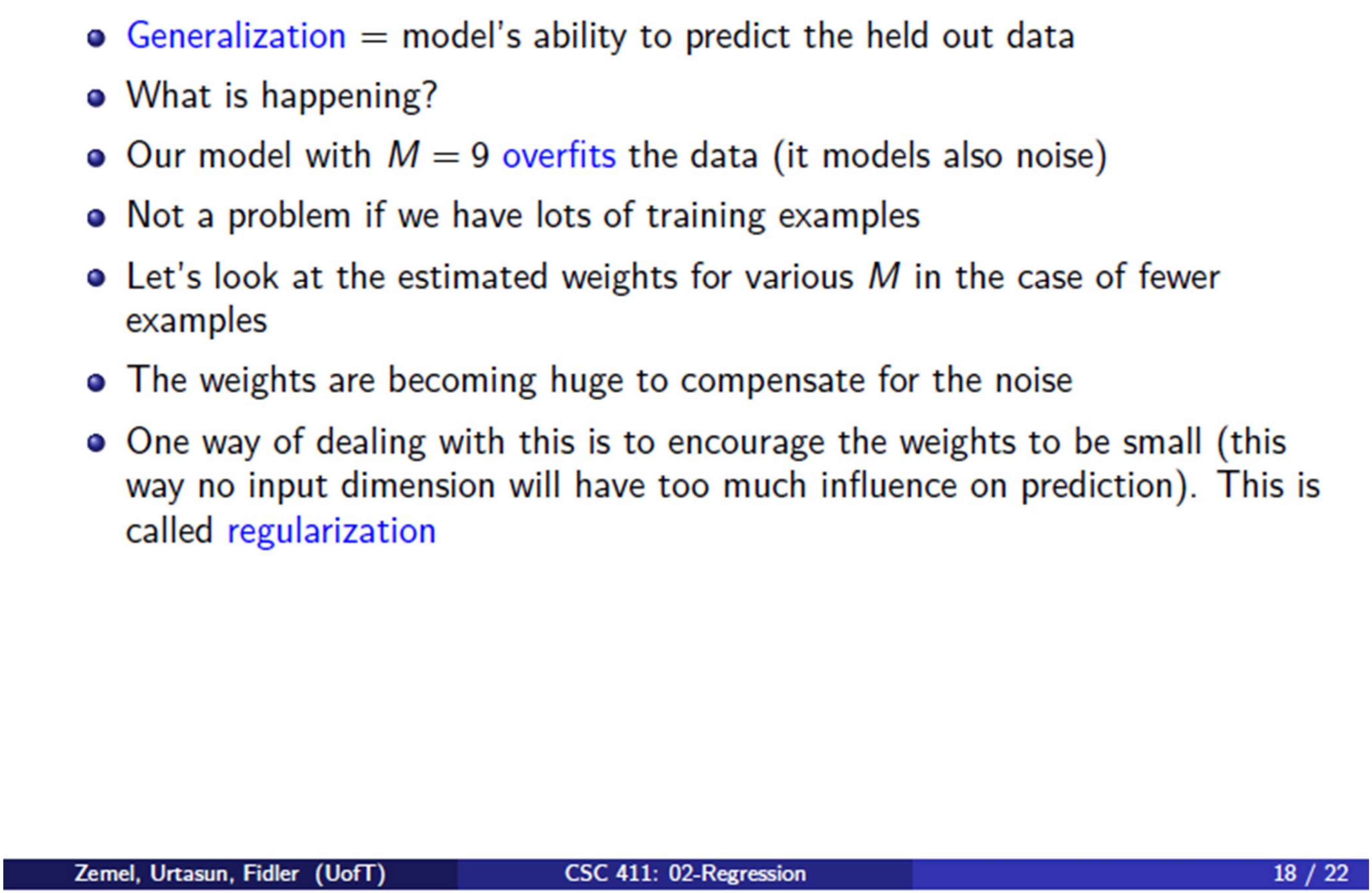


قدرت تعمیم

- ❑ بیش برازش، برازش مناسب، برازش ناکافی



قدرت تعمیم



- Generalization = model's ability to predict the held out data
- What is happening?
- Our model with $M = 9$ overfits the data (it models also noise)
- Not a problem if we have lots of training examples
- Let's look at the estimated weights for various M in the case of fewer examples
- The weights are becoming huge to compensate for the noise
- One way of dealing with this is to encourage the weights to be small (this way no input dimension will have too much influence on prediction). This is called regularization

قدرت تعمیم

- ❑ افزایش بی رویه وزنها در بیش برازش

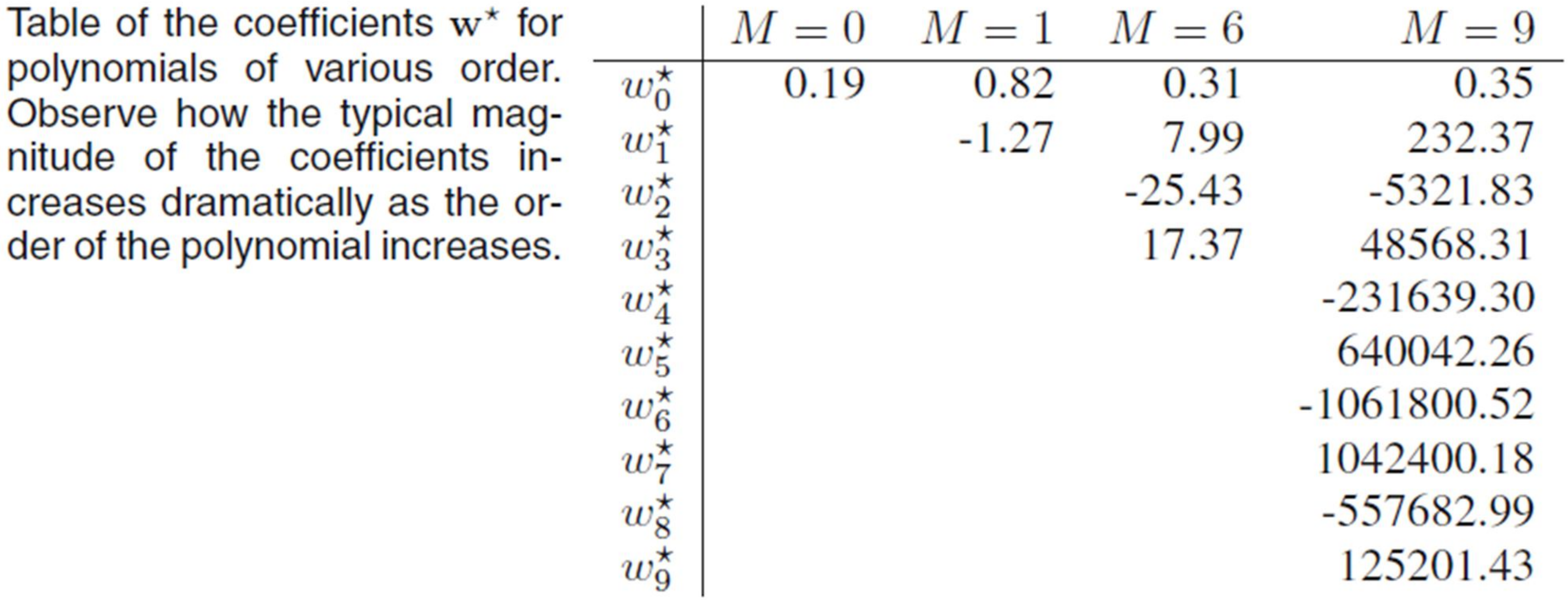


Table of the coefficients $\mathbf{w}^{\star}$ for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
$w_0^{\star}$	0.19	0.82	0.31	0.35
$w_1^{\star}$		-1.27	7.99	232.37
$w_2^{\star}$			-25.43	-5321.83
$w_3^{\star}$			17.37	48568.31
$w_4^{\star}$				-231639.30
$w_5^{\star}$				640042.26
$w_6^{\star}$				-1061800.52
$w_7^{\star}$				1042400.18
$w_8^{\star}$				-557682.99
$w_9^{\star}$				125201.43

کمترین مربعات تنظیم شده

- Increasing the input features this way can complicate the model considerably
- Goal: select the appropriate model complexity automatically
- Standard approach: regularization

$$\tilde{\ell}(\mathbf{w}) = \sum_{n=1}^{N} [t^{(n)} - (w_0 + w_1 x^{(n)})]^2 + \alpha \mathbf{w}^T \mathbf{w}$$

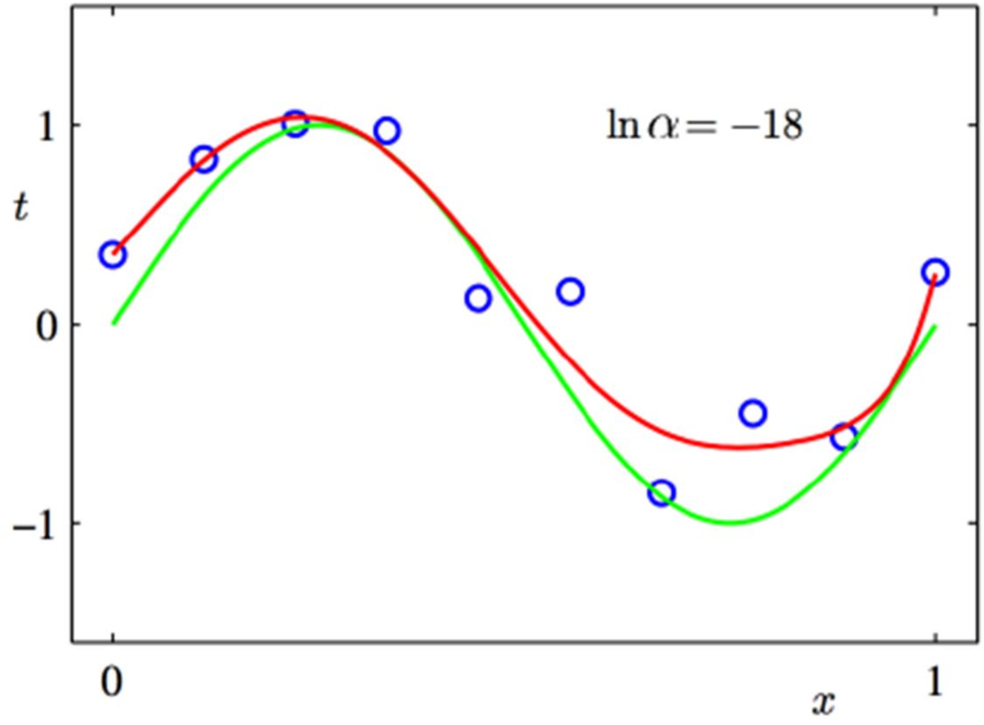
- Intuition: Since we are minimizing the loss, the second term will encourage smaller values in $\mathbf{w}$
- When we use the penalty on the squared weights we have ridge regression in statistics
- Leads to a modified update rule for gradient descent:

$$\mathbf{w} \leftarrow \mathbf{w} + 2\lambda[\sum_{n=1}^{N} (t^{(n)} - y(x^{(n)}))x^{(n)} - \alpha \mathbf{w}]$$

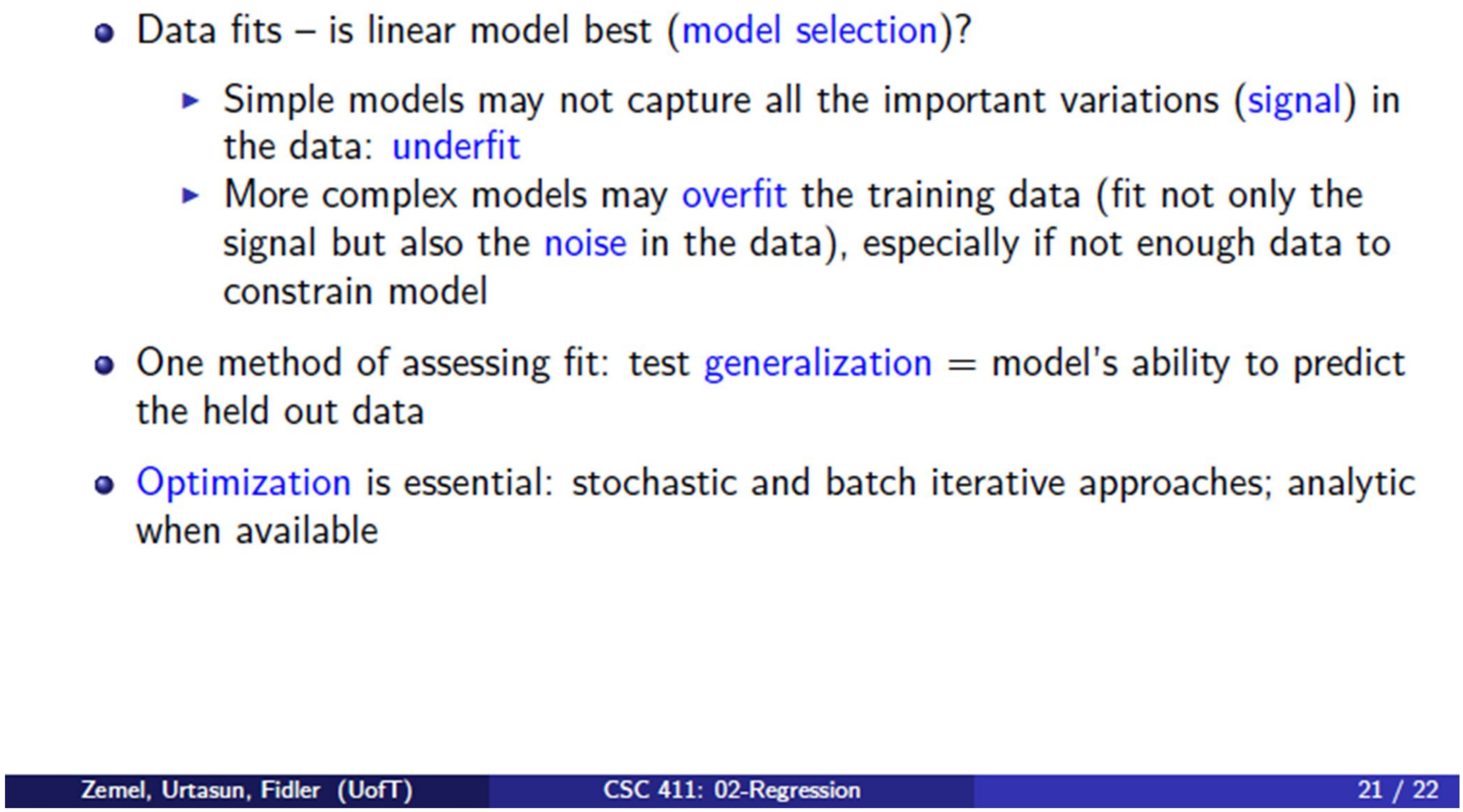
- Also has an analytical solution: $\mathbf{w} = (\mathbf{X}^T\mathbf{X} + \alpha\, \mathbf{I})^{-1}\mathbf{X}^T\mathbf{t}$ (verify!)

کمترین مربعات تنظیم شده

- Better generalization
- Choose α carefully



مرور مفاهیم کلیدی



- Data fits – is linear model best (model selection)?
  - Simple models may not capture all the important variations (signal) in the data: underfit
  - More complex models may overfit the training data (fit not only the signal but also the noise in the data), especially if not enough data to constrain model
- One method of assessing fit: test generalization = model's ability to predict the held out data
- Optimization is essential: stochastic and batch iterative approaches; analytic when available