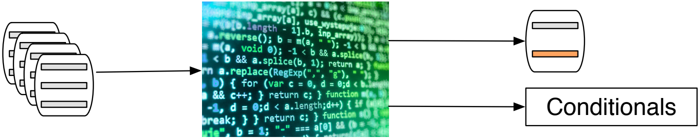
Machine Learning
Regression

Rajesh Ranganath

Machine Learning



$$p(y \mid \mathbf{x})$$

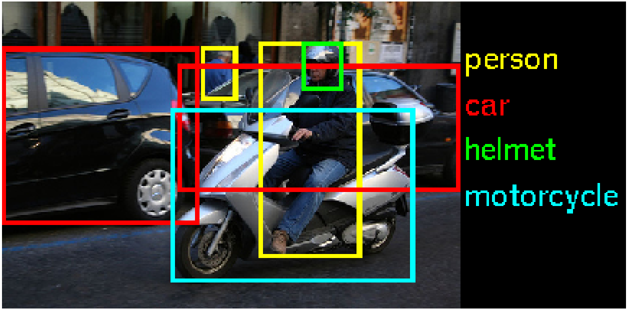
[Image of code from Atlantic]

Supervised Learning

Goal: Learn how to predict y from $\mathbf{x}$

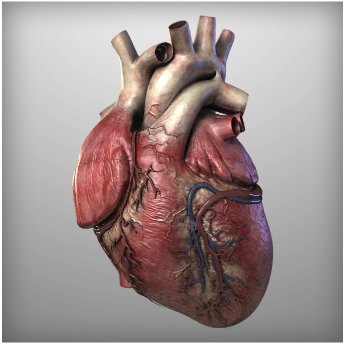
[Mnist, Wikipedia]

Goal: Learn how to predict y from $\mathbf{x}$



[Imagenet]

Goal: Learn how to predict y from $\mathbf{x}$



Supervised Learning

- $\mathbf{x}$ are features or covariates
- y is a real number to be predicted

Observe n samples

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$$

Supervised Learning

Use housing prices as a running example

- **x** are feature or covariates
  - Number of rooms
  - Square feet
  - Zip code
  - Number of bathrooms
- **y** is a real number to be predicted
  - The price of a house

Supervised Learning

How do we build a model?

Start with a function that takes in $\mathbf{x}$

$$f_{\boldsymbol{\theta}}(\mathbf{x})$$

Make distance between $f_{\boldsymbol{\theta}}(\mathbf{x})$ and y small

$$\min_{\boldsymbol{\theta}} \text{distance}(f_{\boldsymbol{\theta}}(\mathbf{x}), y) \triangleq \min_{\boldsymbol{\theta}} d(f_{\boldsymbol{\theta}}(\mathbf{x}), y)$$

Distance examples

- Squared Error

$$d(a, b) = (a - b)^2$$

- Absolute Error

$$d(a, b) = |a - b|$$

Supervised Learning

How do we build a model?

Start with a function that takes in $\mathbf{x}$

$$f_{\boldsymbol{\theta}}(\mathbf{x})$$

Make distance between $f_{\boldsymbol{\theta}}(\mathbf{x})$ and y small

$$\min_{\boldsymbol{\theta}} \text{distance}(f_{\boldsymbol{\theta}}(\mathbf{x}), y) \triangleq \min_{\boldsymbol{\theta}} d(f_{\boldsymbol{\theta}}(\mathbf{x}), y)$$

Do it on the training data of (n) points

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} d(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i)$$

Supervised Learning

Learning a model:

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} d(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i)$$

Make Predictions on a new point $\mathbf{x}^*$:

$$\hat{y} = f_{\boldsymbol{\theta}}(\mathbf{x}^*)$$

Evaluate:

$$\text{error} = d(y^*, \hat{y}).$$

Is there something wrong?

Supervised Learning

Learning a model:

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} d(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) \triangleq \mathscr{L}$$

What happens if f is really flexible?

Supervised Learning

Learning a model:

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} d(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) \triangleq \mathscr{L}$$

What happens if f is really flexible?

$$f_{\boldsymbol{\theta}}^{\text{best}}(\mathbf{x}_i) = y_i$$

The objective is $\mathscr{L}$ zero. On a test point $\mathbf{x}^*$

$$f_{\boldsymbol{\theta}}^{\text{best}}(\mathbf{x}_i) = ????$$

$$f_{\boldsymbol{\theta}}^{\text{best}}(\mathbf{x}_i) = ????$$

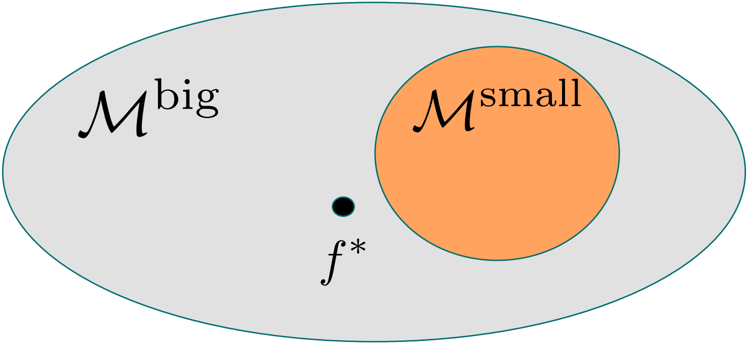
Predictions:

- On a $\mathbf{x}_i$ equals y_i
- On a $\mathbf{x}^* \neq \mathbf{x}_i$ is arbitrary

Can be arbitrary! Doesn't really work. What happened?

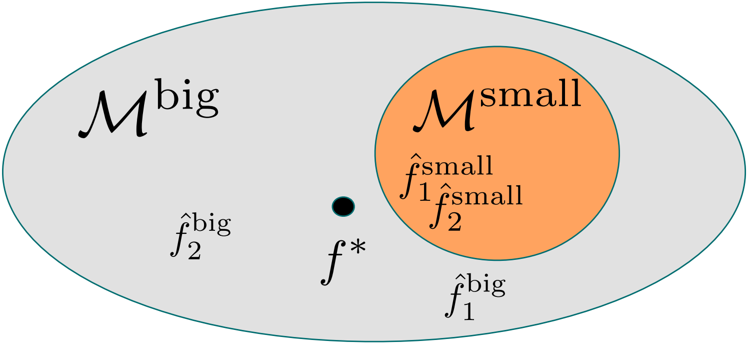
- Best one can do without assumptions
- Overfit to the training data
- Training data has finite size

Overfitting vs Underfitting



Overfitting vs Underfitting

Get two house price data sets $\mathscr{D}^1$, $\mathscr{D}^2$



More later

Linear Regression

Model: linear functions

$$f_{\boldsymbol{\theta}}(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x}$$

- $\mathbf{x}$: p dimensional vector of features (house age, square feet, number of rooms)
- y: house price
- $\boldsymbol{\theta}$: p dimensional regression coefficients

Linear Regression

Model: linear functions

$$f_{\boldsymbol{\theta}}(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x}$$

Distance: Squared Error

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} d(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i)$$

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

- Intercepts handled by including a column of 1 in $\mathbf{x}$

Linear Regression: How to Solve

Goal:

$$\min_{\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

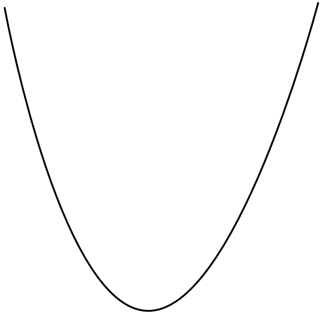
How do we optimize?

Linear Regression: How to Solve

Goal:

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

How do we optimize?



Derivative Zero at Critical Point

Take derivative and set it to zero!

Linear Regression: How to Solve

Minimize:

$$\mathscr{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Differentiate:

$$\nabla_{\boldsymbol{\theta}}\mathscr{L} \quad = \quad \nabla_{\boldsymbol{\theta}}\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Differentiate:

$$\begin{aligned} \nabla_{\boldsymbol{\theta}}\mathcal{L} &= \nabla_{\boldsymbol{\theta}}\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 \end{aligned}$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Differentiate:

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}}\mathcal{L} &= \nabla_{\boldsymbol{\theta}}\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)
\end{aligned}$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Differentiate:

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}}\mathcal{L} &= \nabla_{\boldsymbol{\theta}}\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i) \\
&= \frac{1}{n}\sum_{i=1}^{n}2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i
\end{aligned}$$

Linear Regression: How to Solve

Minimize:

$$\mathscr{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\tfrac{1}{n}\textstyle\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0$$

Linear Regression: How to Solve

Minimize:

$$\mathscr{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\frac{1}{n}\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0$$
$$\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\begin{gathered}
\tfrac{1}{n}\textstyle\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - y_i\mathbf{x}_i = 0
\end{gathered}$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\begin{gathered}
\tfrac{1}{n}\textstyle\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - y_i\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0
\end{gathered}$$

Linear Regression: How to Solve

Minimize:

$$\mathscr{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\begin{gathered}
\tfrac{1}{n}\textstyle\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - y_i\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - \textstyle\sum_{i=1}^{n} y_i\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n} \mathbf{x}_i(\mathbf{x}_i^\top \boldsymbol{\theta}) - \textstyle\sum_{i=1}^{n} y_i\mathbf{x}_i = 0
\end{gathered}$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\begin{gathered}
\tfrac{1}{n}\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - y_i\mathbf{x}_i = 0 \\
\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0 \\
\sum_{i=1}^{n} \mathbf{x}_i(\mathbf{x}_i^\top \boldsymbol{\theta}) - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0 \\
\left(\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^\top\right)\boldsymbol{\theta} - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0
\end{gathered}$$

Linear Regression: How to Solve

Minimize:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Set Derivative to Zero:

$$\begin{gathered}
\tfrac{1}{n}\textstyle\sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - y_i\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0 \\
\textstyle\sum_{i=1}^{n} \mathbf{x}_i(\mathbf{x}_i^\top \boldsymbol{\theta}) - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0 \\
\left(\textstyle\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^\top\right)\boldsymbol{\theta} - \sum_{i=1}^{n} y_i\mathbf{x}_i = 0 \\
\boldsymbol{\theta} = \left(\textstyle\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^\top\right)^{-1}\sum_{i=1}^{n} y_i\mathbf{x}_i
\end{gathered}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$\mathbf{x}_i : (p \times 1) \text{ vector}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$\mathbf{x}_i \mathbf{x}_i^\top : (p \times p) \text{ matrix}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$\mathbf{x}_i \mathbf{x}_i^\top : (p \times p) \text{ matrix}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$\left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} : (p \times p) \text{ matrix}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$y_i : \text{ scalar}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$y_i \mathbf{x}_i : p \times 1 \text{ vector}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$\sum_{i=1}^{n} y_i \mathbf{x}_i : p \times 1 \text{ vector}$$

Linear Regression: Do the dimensions work out?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Sizes:

$$\left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i : p \times 1 \text{ vector}$$

Can we rewrite the solution using linear algebra?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Define

- Matrix $\mathbf{X}$: $n \times p$ matrix
  - Each row is a training example
  - Each column is collection of features

Can we rewrite the solution using linear algebra?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

Define

- Matrix $\mathbf{X}$: $n \times p$ matrix
  - Each row is a training example
  - Each column is collection of features
- Vector $\mathbf{y}$: $n \times 1$ vector
  - Labels for each training example

Can we rewrite the solution using linear algebra?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

With $\mathbf{X}$ and $\mathbf{y}$

$$\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top = \mathbf{X}^\top \mathbf{X}$$

Can we rewrite the solution using linear algebra?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

With $\mathbf{X}$ and $\mathbf{y}$

$$\sum_{i=1}^{n} y_i \mathbf{x}_i = \mathbf{X}^\top \mathbf{y}$$

Can we rewrite the solution using linear algebra?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

With $\mathbf{X}$ and $\mathbf{y}$

$$\boldsymbol{\theta}^* = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{y}$$

Can we rewrite the solution using linear algebra?

Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

With $\mathbf{X}$ and $\mathbf{y}$

$$\boldsymbol{\theta}^* = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{y}$$

Can derive directly with matrix calculus!

Linear Regression: How long does it take?

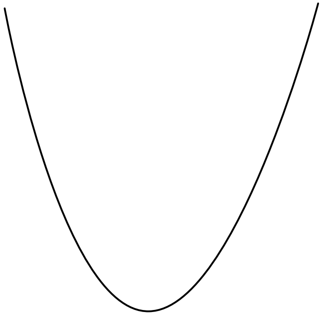
Optimal regression coefficients

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

- $O(n)$ for number of examples
- $O(p^3)$ for matrix inversion

Can be slow if $O(n)$ is too big or if p is too big

An other way to optimize



Derivative Zero at Critical Point

?

Gradient Points in Steepest Direction

Local function change along direction $\mathbf{u}$ with norm 1

$$D_{\mathbf{u}}(\boldsymbol{\theta})[\mathscr{L}] = \lim_{h \to 0} \frac{\mathscr{L}(\boldsymbol{\theta} + h\mathbf{u}) - \mathscr{L}(\boldsymbol{\theta})}{h}$$

Assume $\mathscr{L}$ is differentiable

$$\begin{aligned} D_{\mathbf{u}}(\boldsymbol{\theta})[\mathscr{L}] &= \lim_{h \to 0} \frac{\mathscr{L}(\boldsymbol{\theta}) + \nabla \mathscr{L}(\boldsymbol{\theta})^\top (\boldsymbol{\theta} + h\mathbf{u} - \boldsymbol{\theta}) + o(|\boldsymbol{\theta} + h\mathbf{u} - \boldsymbol{\theta}|) - \mathscr{L}(\boldsymbol{\theta})}{h} \\ &= \frac{\nabla \mathscr{L}(\boldsymbol{\theta})^\top (h\mathbf{u}) + o(|\boldsymbol{\theta} + h\mathbf{u} - \boldsymbol{\theta}|)}{h} \\ &= \nabla \mathscr{L}(\boldsymbol{\theta})^\top (\mathbf{u}) \end{aligned}$$

Gradient Points in Steepest Direction

Local function change along direction $\mathbf{u}$ with norm 1

$$D_{\mathbf{u}}(\boldsymbol{\theta})[\mathcal{L}] = \lim_{h \to 0} \frac{\mathcal{L}(\boldsymbol{\theta} + h\mathbf{u}) - \mathcal{L}(\boldsymbol{\theta})}{h}$$

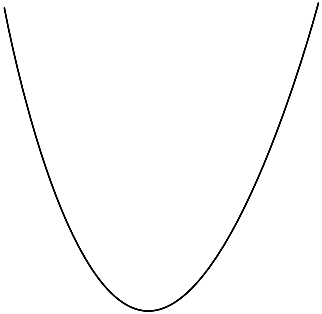
Assume $\mathcal{L}$ is differentiable

$$\begin{aligned} D_{\mathbf{u}}(\boldsymbol{\theta})[\mathcal{L}] &= \lim_{h \to 0} \frac{\mathcal{L}(\boldsymbol{\theta}) + \nabla \mathcal{L}(\boldsymbol{\theta})^\top (\boldsymbol{\theta} + h\mathbf{u} - \boldsymbol{\theta}) + o(|\boldsymbol{\theta} + h\mathbf{u} - \boldsymbol{\theta}|) - \mathcal{L}(\boldsymbol{\theta})}{h} \\ &= \frac{\nabla \mathcal{L}(\boldsymbol{\theta})^\top (h\mathbf{u}) + o(|\boldsymbol{\theta} + h\mathbf{u} - \boldsymbol{\theta}|)}{h} \\ &= \nabla \mathcal{L}(\boldsymbol{\theta})^\top (\mathbf{u}) \end{aligned}$$

$D_{\mathbf{u}}(\boldsymbol{\theta})[\mathcal{L}] = \nabla \mathcal{L}(\boldsymbol{\theta})^\top (\mathbf{u})$ maximized when $\mathbf{u} = \frac{\nabla \mathcal{L}(\boldsymbol{\theta})}{||\nabla \mathcal{L}(\boldsymbol{\theta})||}$

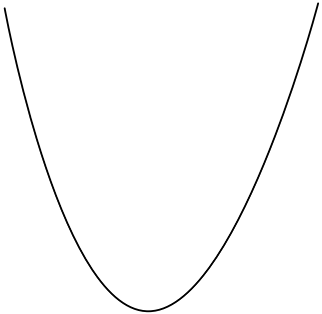
Gradient is steepest ascent direction

An other way to optimize



Derivative Zero at Critical Point

An other way to optimize



Derivative Zero at Critical Point

Follow negative gradient

Gradient Descent

1. Start with initial parameters $\boldsymbol{\theta}_0$
2. Step size or learning rate ρ_t
3. Update:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \rho_t \nabla_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta}_{t-1})$$

Intuitively minimizes a linear approximation of the function locally

Gradient Descent Time Complexity

$$\nabla_{\boldsymbol{\theta}} \mathscr{L} = \frac{1}{n} \sum_{i=1}^{n} 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i$$

- Each gradient computation takes time $O(np)$
- May need many steps

Slow for large number of training examples

Linear Regression: How to Solve With Big Data

Minimize:

$$\mathscr{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Make a distribution F

$$F = \frac{1}{n}\sum_{i=1}^{n}\delta_{(\mathbf{x}_i, y_i)}$$

Linear Regression: How to Solve With Big Data

Minimize:

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2$$

Make a distribution F

$$F = \frac{1}{n}\sum_{i=1}^{n}\delta_{(\mathbf{x}_i, y_i)}$$

Objective is an expectation

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 = \mathbb{E}_{(\mathbf{x}_i, y_i)\sim F}[(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2]$$

Maybe we can optimize using samples from F

Stochastic Optimization

All about optimizing functions that are expectations.

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{x \sim F}[f(x, \boldsymbol{\theta})].$$

Can differentiate

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{x \sim F}[\nabla_{\boldsymbol{\theta}} f(x, \boldsymbol{\theta})].$$

Can get a noisy gradient by sampling from F

$$\hat{\nabla}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} f(x_i, \boldsymbol{\theta}) \text{ where } x_i \sim F$$

Fast to compute and unbiased:

$$\mathbb{E}_F[\hat{\nabla}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})] = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$$

Stochastic Optimization

Can we use noisy, unbiased gradients $\hat{\nabla}_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta})$ to optimize?

Stochastic Optimization

Can we use noisy, unbiased gradients $\hat{\nabla}_{\boldsymbol{\theta}}\mathscr{L}(\boldsymbol{\theta})$ to optimize? Yes!

Stochastic Optimization

Can we use noisy, unbiased gradients $\hat{\nabla}_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta})$ to optimize? Yes!

1. Start with initial parameters $\boldsymbol{\theta}_0$
2. Step size or learning rate ρ_t
3. Sample data point $\mathbf{x}_i \sim F$
4. Update:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \rho_t \hat{\nabla}_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta}_{t-1})(\mathbf{x}_i)$$

Stochastic Optimization: Why does it work?

- Need conditions on step sizes
- Not summable

$$\sum_{t=1}^{\infty} \rho_t = \infty$$

- Square summable

$$\sum_{t=1}^{\infty} \rho_t^2 < \infty$$

Intuition?

Stochastic Optimization: Why does it work?

- Need conditions on step sizes
- Not summable

$$\sum_{t=1}^{\infty} \rho_t = \infty$$

Reaches far parameters

- Square summable

$$\sum_{t=1}^{\infty} \rho_t^2 < \infty$$

Controls the noise

Linear Regression: Stochastic Optimization

Objective:

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 = \mathbb{E}_{(\mathbf{x}_i,y_i)\sim F}[(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2]$$

Gradient:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}) &= \mathbb{E}_{(\mathbf{x}_i,y_i)\sim F}[\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2] \\ &= \mathbb{E}_{(\mathbf{x}_i,y_i)\sim F}[2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i]\end{aligned}$$

Linear Regression: Stochastic Optimization

Objective:

$$\mathscr{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2 = \mathbb{E}_{(\mathbf{x}_i, y_i)\sim F}[(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2]$$

Gradient:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\mathscr{L}(\boldsymbol{\theta}) &= \mathbb{E}_{(\mathbf{x}_i, y_i)\sim F}[\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)^2] \\ &= \mathbb{E}_{(\mathbf{x}_i, y_i)\sim F}[2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i]\end{aligned}$$

Noisy Gradient

$$\hat{\nabla}\mathscr{L} = 2(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i, \quad (\mathbf{x}_i, y_i) \sim F$$

Linear Regression: Stochastic Optimization

1. Start with initial parameters $\boldsymbol{\theta}_0$
2. Step size or learning rate ρ_t
3. Sample data point $\mathbf{x}_i, y_i \sim F$
4. Update:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - 2\rho_t(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i)\mathbf{x}_i$$

Time complexity per step $O(p)$!

More steps needed due to variance

Stochastic Optimization Summary

- One of the key tools in machine learning AI
- Allows scale to millions of data points
- Later in class we will see more details

Why the squared distance?

Define residuals

$$r_i = y_i - \boldsymbol{\theta}^\top \mathbf{x}_i$$

Define residuals

$$r_i = y_i - \boldsymbol{\theta}^\top \mathbf{x}_i$$

Let's model these residuals probabilistically.

Define residuals

$$r_i = y_i - \boldsymbol{\theta}^\top \mathbf{x}_i$$

Let's model these residuals probabilistically. How do we do that?

$$r_i \sim p$$

Lot’s of possible choices for p

- Gamma
- Gaussian
- Student-T
- A deep model?

Where does the randomness come from?

Where does the randomness come from?

Lots of small effects we couldn't capture

- House color: ϵ_1
- House direction: ϵ_2
- Closeness to freeway: ϵ_3
- Closeness to school: ϵ_4

$$y = f(\mathbf{x}, \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4)$$

Where does the randomness come from?

Lots of small effects we couldn't capture

- House color: ϵ_1
- House direction: ϵ_2
- Closeness to freeway: ϵ_3
- Closeness to school: ϵ_4

$$y = f(\mathbf{x}, \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4)$$

Assume the world is linear

$$y = \boldsymbol{\theta}^\top \mathbf{x} + \sum_i \epsilon_i$$

Where does the randomness come from?

Lots of small effects we couldn't capture

- House color: ϵ_1
- House direction: ϵ_2
- Closeness to freeway: ϵ_3
- Closeness to school: ϵ_4

$$y = f(\mathbf{x}, \epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4)$$

Assume the world is linear

$$y = \boldsymbol{\theta}^\top \mathbf{x} + \sum_i \epsilon_i$$

y and the residuals are random because of missing observations when the model is correct

Suppose a sequence of mean zero random variables $\epsilon_1, ..., \epsilon_n$,
Define

$$s_n^2 = \sum_{i=1}^{n} \mathrm{Var}(\epsilon_i),$$

Suppose a sequence of mean zero random variables $\epsilon_1, \ldots, \epsilon_n$,
Define

$$s_n^2 = \sum_{i=1}^{n} \mathrm{Var}(\epsilon_i),$$

Then if

$$\lim_{n\to\infty} \frac{1}{s_n^{2+\delta}} \sum_{i=1}^{n} \mathbb{E}[|\epsilon_i|^{2+\delta}] = 0,$$

we get

$$\frac{1}{s_n} \sum_{i=1}^{n} \epsilon_i \to \mathrm{Normal}(0, 1)$$

This is the central limit theorem

The central limit theorem suggests the residual should be normal

Residuals

$$r_i = y_i - \boldsymbol{\theta}^\top \mathbf{x}_i \sim \text{Normal}(0, \sigma^2)$$

Implies

$$y_i \sim \text{Normal}(\boldsymbol{\theta}^\top \mathbf{x}_i, \sigma^2)$$

How do we use this to estimate theta?

Residuals

$$r_i = y_i - \boldsymbol{\theta}^\top \mathbf{x}_i \sim \text{Normal}(0, \sigma^2)$$

Implies

$$y_i \sim \text{Normal}(\boldsymbol{\theta}^\top \mathbf{x}_i, \sigma^2)$$

How do we use this to estimate theta?

Maximize the likelihood of the observations

Likelihood of the data

$$p_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) = \prod_{i=1}^{n} p_{\boldsymbol{\theta}}(y_i \mid \mathbf{x}_i)$$

Easier to work the the log-likelihood

$$\log p_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) = \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i \mid \mathbf{x}_i)$$

Likelihood of the data

$$p_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) = \prod_{i=1}^{n} p_{\boldsymbol{\theta}}(y_i \mid \mathbf{x}_i)$$

Easier to work the the log-likelihood

$$\log p_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) = \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(y_i \mid \mathbf{x}_i)$$

To maximize, we

1. Substitute the normal density function

$$p(y_i \mid \mathbf{x}_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2\right)$$

2. Compute gradients

$$
\begin{aligned}
\nabla_{\mathscr{L}} &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log p(y_i \mid \mathbf{x}_i) \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2 \right) \right) \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \left[\log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \right) + \log\left(\exp\left(-\frac{1}{2\sigma^2}(y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2 \right) \right) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \left[-\frac{1}{2\sigma^2}(y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2 \right] \\
&= \sum_{i=1}^{n} \frac{1}{\sigma^2}(y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)\mathbf{x}_i
\end{aligned}
$$

$$
\begin{aligned}
\nabla_{\mathscr{L}} &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log p(y_i \,|\, \mathbf{x}_i) \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} (y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2 \right) \right) \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \left[\log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \right) + \log\left(\exp\left(-\frac{1}{2\sigma^2} (y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2 \right) \right) \right] \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \left[-\frac{1}{2\sigma^2} (y_i - \boldsymbol{\theta}^\top \mathbf{x}_i)^2 \right] \\
&= \sum_{i=1}^{n} \frac{1}{\sigma^2} (y_i - \boldsymbol{\theta}^\top \mathbf{x}_i) \mathbf{x}_i
\end{aligned}
$$

Can set to zero and solve for $\boldsymbol{\theta}^*$

$$
\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i
$$

Does this look familiar?

$$\boldsymbol{\theta}^* = \left(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} y_i \mathbf{x}_i$$

- It's the same solution as linear regression
- Linear regression is maximizes the probability of the data with Normal residuals
- Normal residuals partly justified by central limit theorem

What if f is not linear

$$\min_f \mathbb{E}_{p(\mathbf{x},y)}[(y - f(\mathbf{x}))^2]$$

What if f is not linear

$$\begin{aligned} & \min_f \mathbb{E}_{p(\mathbf{x},y)}[(y - f(\mathbf{x}))^2] \\ = \quad & \min_f \mathbb{E}_{p(\mathbf{x},y)}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \end{aligned}$$

What if f is not linear

$$
\begin{aligned}
& \min_f \mathbb{E}_{p(\mathbf{x},y)}[(y-f(\mathbf{x}))^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x},y)}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})p(y\,|\,\mathbf{x})}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2]
\end{aligned}
$$

What if f is not linear

$$\begin{aligned}
& \min_f \mathbb{E}_{p(\mathbf{x},y)}[(y - f(\mathbf{x}))^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x},y)}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})p(y\,|\,\mathbf{x})}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y^2\,|\,\mathbf{x}] - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right]
\end{aligned}$$

What if f is not linear

$$
\begin{aligned}
& \min_f \mathbb{E}_{p(\mathbf{x},y)}[(y-f(\mathbf{x}))^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x},y)}[y^2-2yf(\mathbf{x})+f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})p(y\,|\,\mathbf{x})}[y^2-2yf(\mathbf{x})+f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y^2\,|\,\mathbf{x}]-2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x})+f(\mathbf{x})^2\right] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y\,|\,\mathbf{x}]^2+\mathbb{V}\mathrm{ar}(y\,|\,\mathbf{x})-2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x})+f(\mathbf{x})^2\right]
\end{aligned}
$$

What if f is not linear

$$
\begin{aligned}
& \min_f \mathbb{E}_{p(\mathbf{x},y)}[(y - f(\mathbf{x}))^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x},y)}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})p(y\,|\,\mathbf{x})}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y^2\,|\,\mathbf{x}] - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y\,|\,\mathbf{x}]^2 + \mathbb{V}\mathrm{ar}(y\,|\,\mathbf{x}) - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y\,|\,\mathbf{x}]^2 - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right] + \mathbb{E}_{p(\mathbf{x})}[\mathbb{V}\mathrm{ar}(y\,|\,\mathbf{x})]
\end{aligned}
$$

What if f is not linear

$$
\begin{aligned}
& \min_f \mathbb{E}_{p(\mathbf{x},y)}[(y-f(\mathbf{x}))^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x},y)}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})p(y\,|\,\mathbf{x})}[y^2 - 2yf(\mathbf{x}) + f(\mathbf{x})^2] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y^2\,|\,\mathbf{x}] - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y\,|\,\mathbf{x}]^2 + \mathbb{V}\mathrm{ar}(y\,|\,\mathbf{x}) - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}[y\,|\,\mathbf{x}]^2 - 2\mathbb{E}[y\,|\,\mathbf{x}]f(\mathbf{x}) + f(\mathbf{x})^2\right] + \mathbb{E}_{p(\mathbf{x})}[\mathbb{V}\mathrm{ar}(y\,|\,\mathbf{x})] \\
= \quad & \min_f \mathbb{E}_{p(\mathbf{x})}\left[(\mathbb{E}[y\,|\,\mathbf{x}] - f(\mathbf{x}))^2\right] + \mathbb{E}_{p(\mathbf{x})}[\mathbb{V}\mathrm{ar}(y\,|\,\mathbf{x})]
\end{aligned}
$$

The optimal f is the conditional expectation

- What happens if there are too many features?
  Linear regression overfits again

- What if the outcome variable is binary?