

Lecture 1

Introduction to Modeling

DSC 40A, Summer 2024

Agenda

- Introductions.
- What is DSC 40A about?
- Logistics.
- Modeling.
- The constant model.

Introductions

Instructor: Nishant Kheterpal

- Originally from Ann Arbor, MI, .
- Undergrad: EECS at Berkeley.
- Grad school: MS, PhD-in-progress in Robotics at Michigan.
- Teaching this summer in the Halıcıoğlu Data Science Institute at UC San Diego.
 - Previously taught DSC 10 during the first summer session.
- Helped teach other classes at Michigan and Berkeley.
- Outside interests: traveling, cooking, baking, working on bicycles.



My summer so far.

Course staff

We have 3 tutors, all of whom are excited to help you in discussion and office hours!

Jack Determan

Zoe Ludena

Owen Miller

Read more about us at dsc40a.com/staff.

Throughout lecture, ask questions!

- You're always free to ask questions during lecture, and I'll try and stop for them frequently. But still, you may not feel like asking your question out loud.
- You can **type your questions anonymously** at the following link and I'll try and answer them.

q.dsc40a.com

- You'll also use this form to answer questions that I ask you during lecture.
- If the direct link doesn't work, use the 🤔 **Lecture Questions** link in the top right corner of dsc40a.com.



Ed



Gradescope



Practice



Lecture Questions

Question 🤔

Answer at q.dsc40a.com

Select the **FALSE** statement below.

- A: I've been making sourdough bread for four years.
- B: I've been to Japan four times.
- C: I skipped first grade.
- D: I've tried out for Jeopardy four times.
- E: I am less than 28 years old.

What is DSC 40A about?

Theoretical Foundations of Data Science I

Mathematical Foundations of Machine Learning

What have you *heard* about DSC 40A?

Here are some responses from the Welcome Survey in the spring quarter.

I've heard the class seeks to uncover a lot of the key concepts of the math behind machine learning, while utilizing a lot of linear algebra. I've heard that the class can be difficult and proof-heavy.

I heard it is conceptual, and therefore, a pretty hard class (to understand conceptually). I also heard it has a lot to do with linear algebra.

That it's the most awful class in the DSC major, pretty much just pure math/all proofs.

It's a pretty hard class but rewarding in the end.

Why do we need to study theoretical foundations?

MNIST

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Machine learning is about automatically learning patterns from data.

Humans are good at understanding handwriting – but how do we get computers to understand handwriting?

Course overview

Part 1: Learning from Data (Weeks 1, 2, and 3)

- Summary statistics and loss functions; empirical risk minimization.
- Linear regression (including multiple variables); linear algebra.
- Clustering.

Mid term

Part 2: Probability (Weeks 4 and 5)

- Set theory and combinatorics; probability fundamentals.
- Conditional probability and independence.
- The Naïve Bayes classifier.

Final

→ backbone of ML-type optimization

Learning objectives

After this class, you'll...

- understand the basic principles underlying almost every machine learning and data science method.
- be better prepared for the math in upper division: vector calculus, linear algebra, and probability.

What do DSC 80 students have to say about DSC 40A?

Here are some responses from the End-of-Quarter Survey last ^{winter} quarter in DSC 80.

study hardy, pay attention in DSC 40A and start work early :)

40A and Math 18 is super important for this class. Don't wait till the last minute too!

I think DSC40[A] was the most important prerequisite for this class.

Logistics

Getting started

- The course website, dsc40a.com, contains all content. **Read the syllabus carefully!**
 - Click around; you'll find other helpful resources.
- Other sites you'll need to use:
 - [Gradescope](#) is where you'll submit all assignments. You'll be automatically added within 24 hours of enrolling.
 - [Ed](#) is where all announcements will be made. If you're not enrolled, there's a join link in the syllabus.
 - We aren't using Canvas.
- Make sure to fill out the [Welcome Survey](#) ASAP.

Lectures

- Lecture is here, WLH 2208, Monday-Thursday 12:30-1:50p.
- Lecture slides will be posted on the course website before class, and annotated slides will be posted after class.
- Lecture will be podcasted. → automatically on podcast. ucsd.edu
- The value of lecture is interaction and discussion, so even though attendance isn't required, it's highly, highly recommended.

Use these to study

Discussions

- Discussion weekly on Wednesdays, directly after lecture here in WLH 2208, 2-3:50p.
- Discussion will primarily be used for **groupwork** – that is, working on problems in small groups of size 2-4.
 - If you email me, you may work in a self-organized group outside of a discussion section for full credit, but no matter what, **you cannot work alone**.
- Groupwork worksheets are due to Gradescope on **Mondays at 11:59p**.
 - Only one group member needs to submit, and should add the rest of the group to the submission.
- **The value of attending is getting support from tutors and working in a group.**

Grading

- **Homeworks (40%)**: Due to Gradescope **Tuesdays and Fridays at 11:59p**, due dates vary.
 - Graded for correctness. Lowest score is dropped. ?? *subject to change*
- **Groupworks (10%)**: Due to Gradescope on ~~Mondays~~ at 11:59p.
 - Graded for effort. Lowest score is dropped.
- **Midterm Exam (20%)**: Thursday, August 22nd, in class.
- **Final Exam (30%)**: Friday, September 6th, 11:30a-2:30p in WLH 2208. See the [syllabus](#) for the redemption policy.

Let us know about exam conflicts on the [Welcome Survey](#).

Support

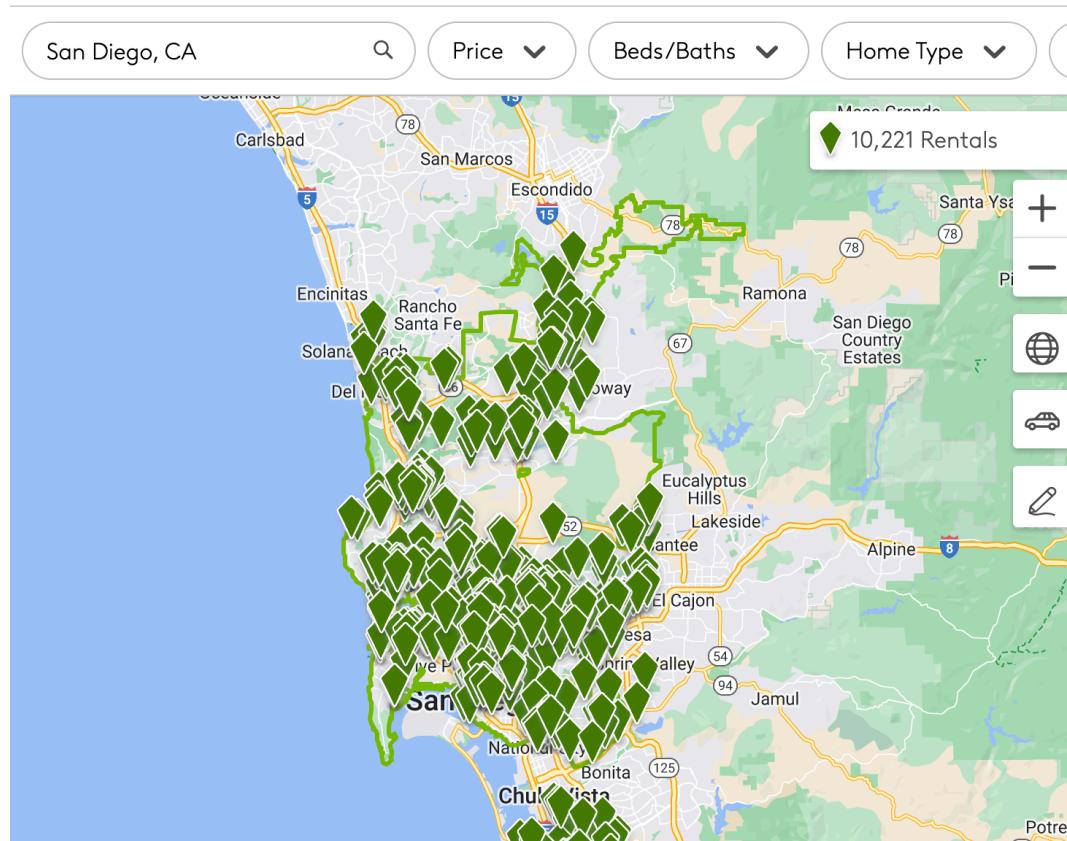
We know this is a challenging class, and we're here to help:

- **Office hours:** In-person in HDSI 155 and virtual on [Zoom](#). Plan to attend at least twice a week for homework help.
- **Ed:** Use it! We're here to help you. Post conceptual questions publicly – just don't post answers to homework questions.

A bunch of new-ish things to improve the student experience:

- [practice.dsc40a.com](#) to give you access to practice exam problems, categorized by topic.
- Walkthrough videos to show you our thought process when answering questions.
- More time reviewing linear algebra.

Modeling



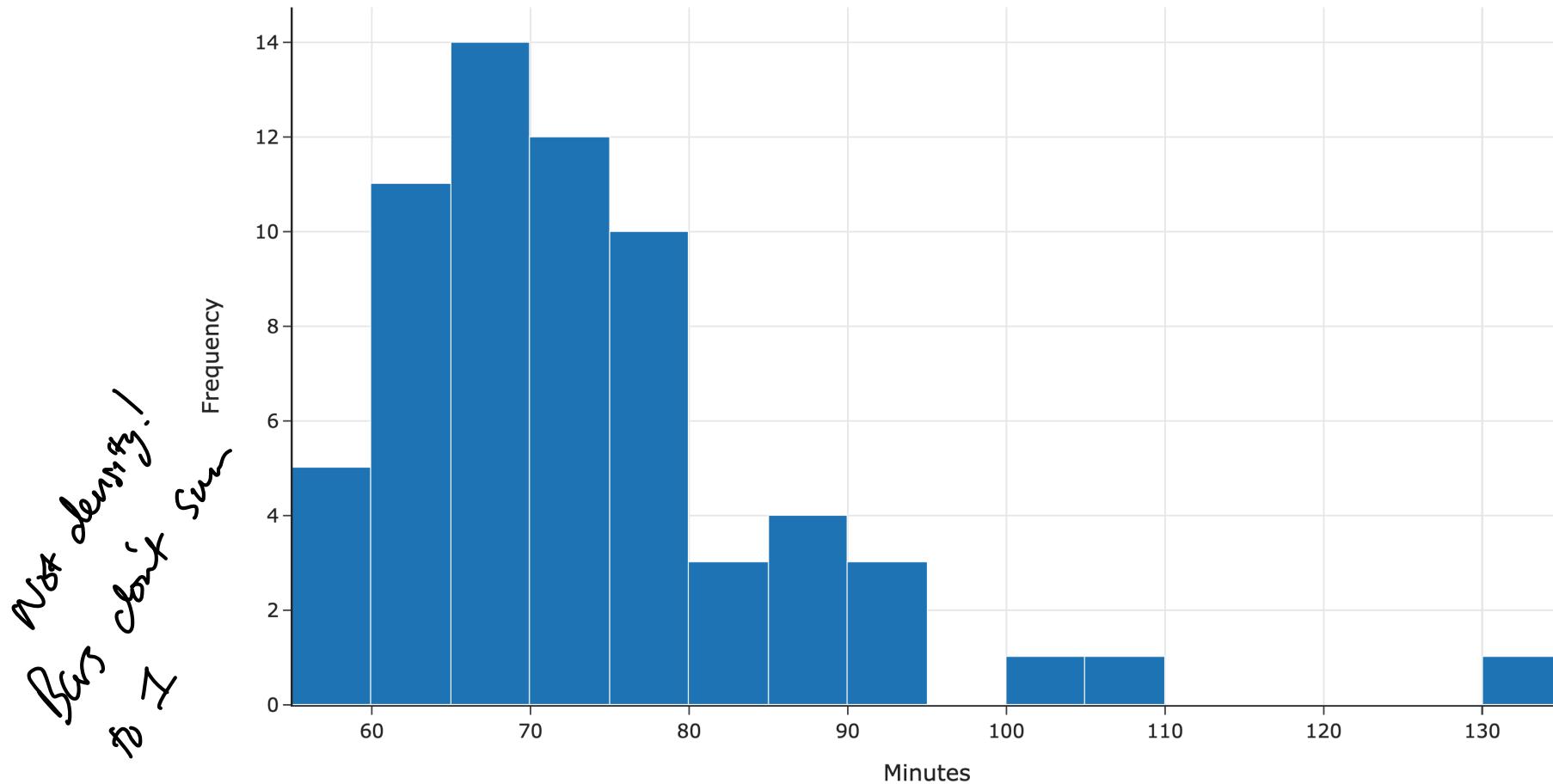
You might be starting to look for off-campus apartments, none of which are affordable.

	date	day	departure_hour	minutes
0	5/22/2023	Mon	8.450000	63.0
1	9/18/2023	Mon	7.950000	75.0
2	10/17/2023	Tue	10.466667	59.0
3	11/28/2023	Tue	8.900000	89.0
4	2/15/2024	Thu	8.083333	69.0

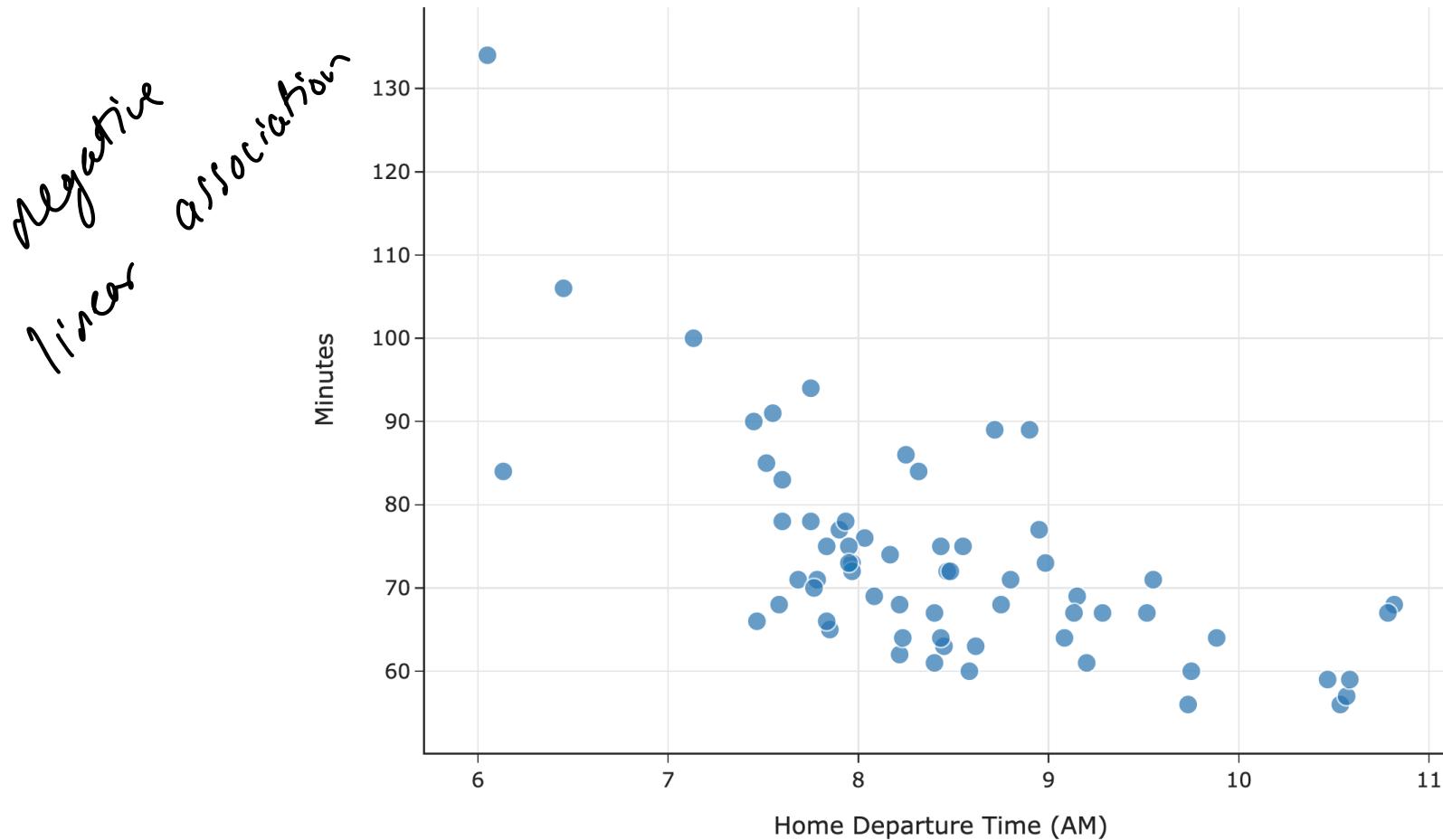
...

You decide to live with your parents in Orange County and commute.
You keep track of how long it takes you to get to school each day.

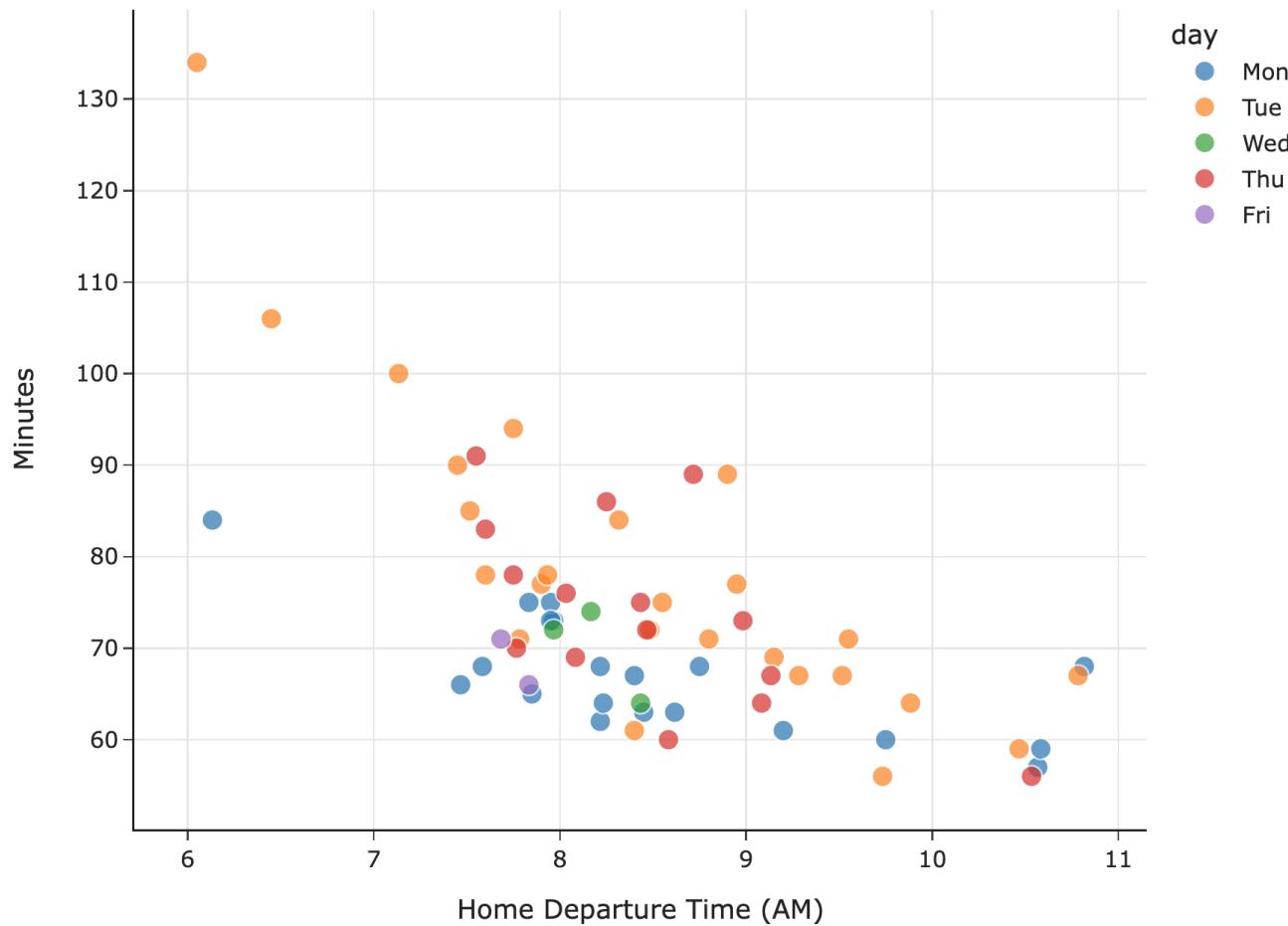
Distribution of Commuting Time



Commuting Time vs. Home Departure Time



Commuting Time vs. Home Departure Time



Goal: Predict your commute time.

That is, predict how long it'll take to get to school.

data in future looks like data
in the past

How can we do this? learn a pattern
from data

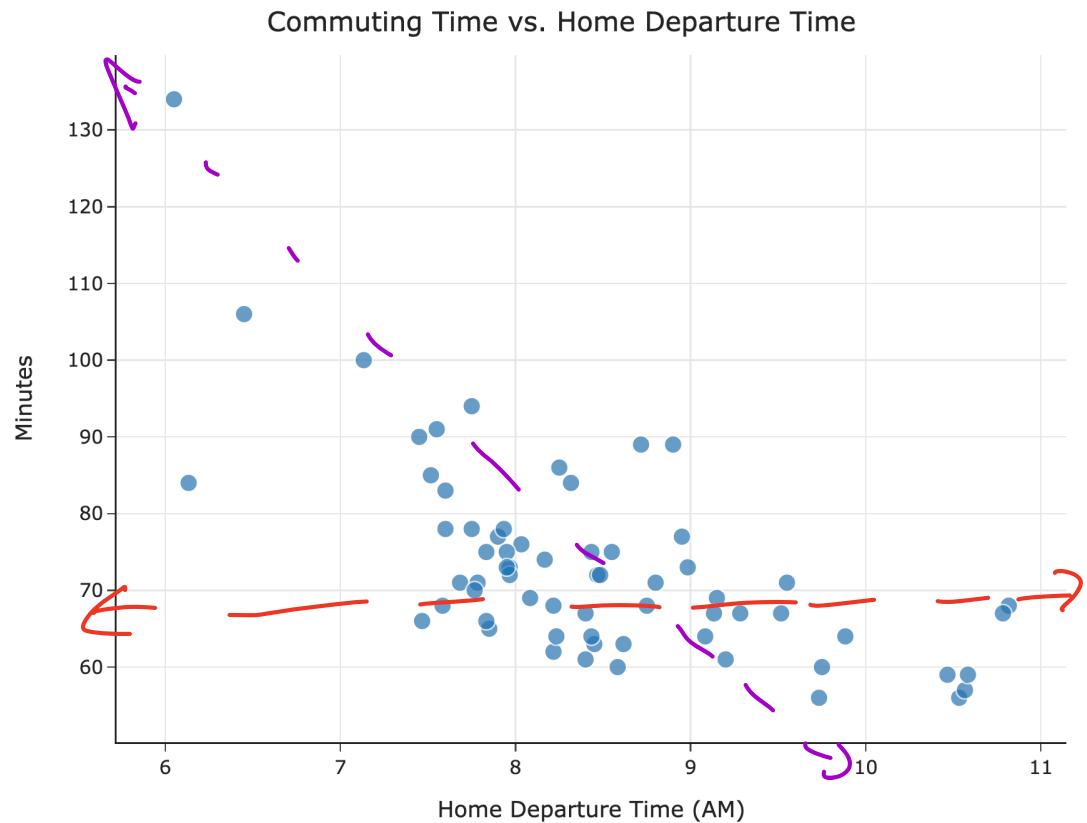
What will we need to assume?

"*datum*" singular

A **model** is a set of assumptions about how data were generated.

Was?

Possible models



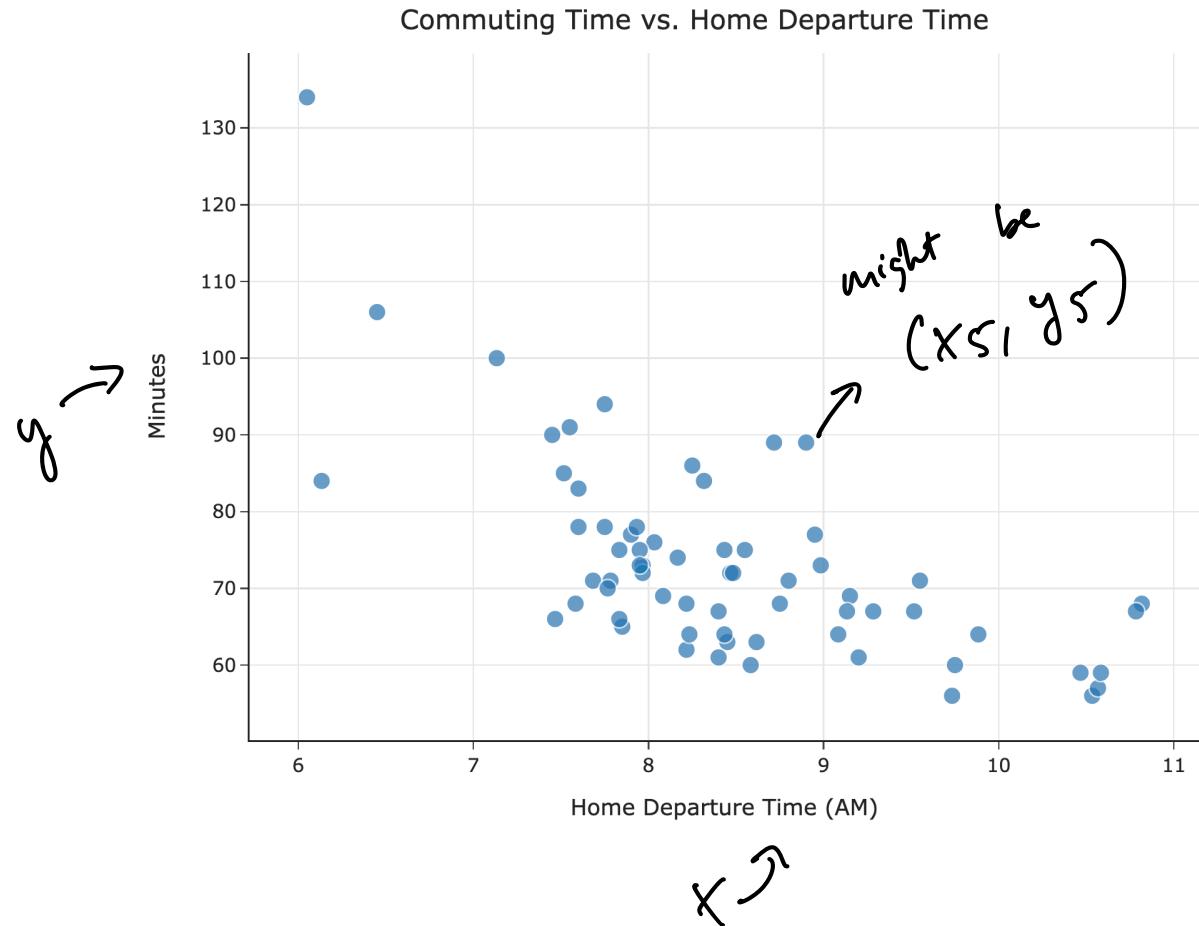
simple linear regression

model

constant model

↳ Summary statistics

Notation



x : "input", "independent variable", or "feature"

y : "response", "dependent variable", or "target"

We use x to predict y .

The i th observation is denoted (x_i, y_i) .

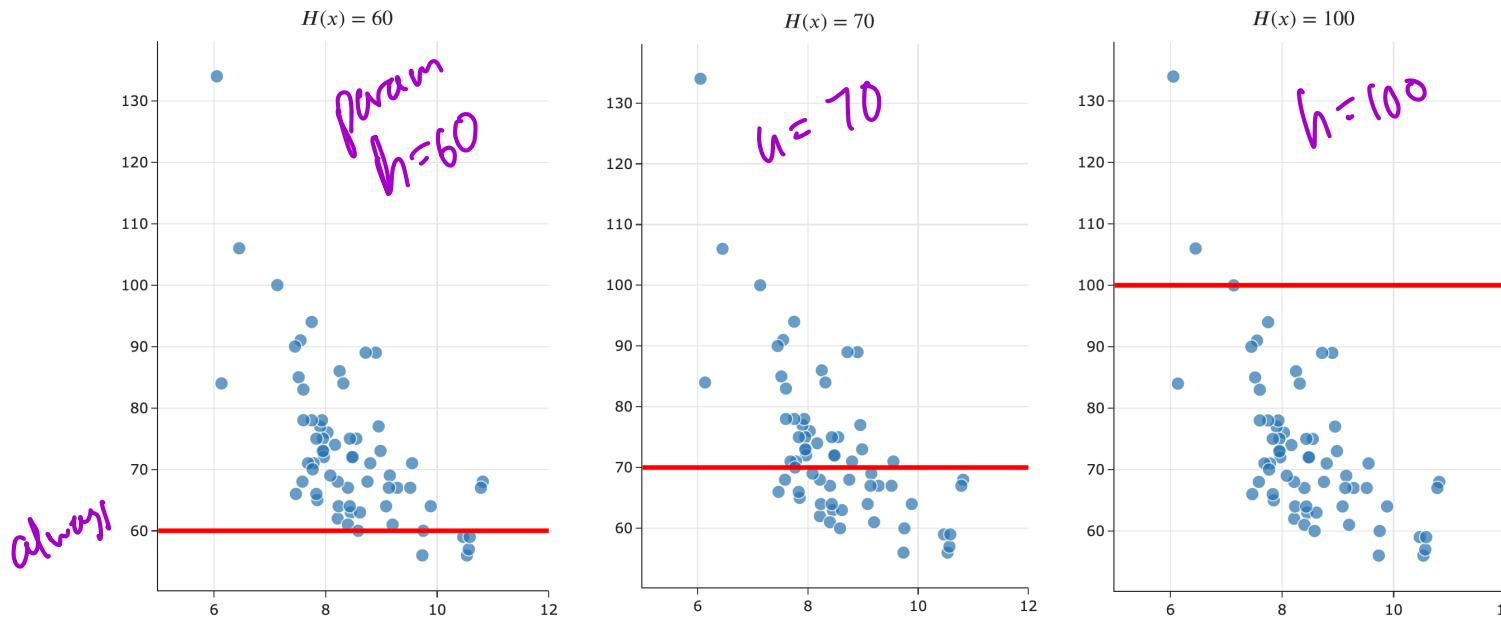
Hypothesis functions and parameters

use H to make predictions

A hypothesis function, H , takes in an x as input and returns a predicted y .

Parameters define the relationship between the input and output of a hypothesis function.

The constant model, $H(x) = h$, has one parameter: h .

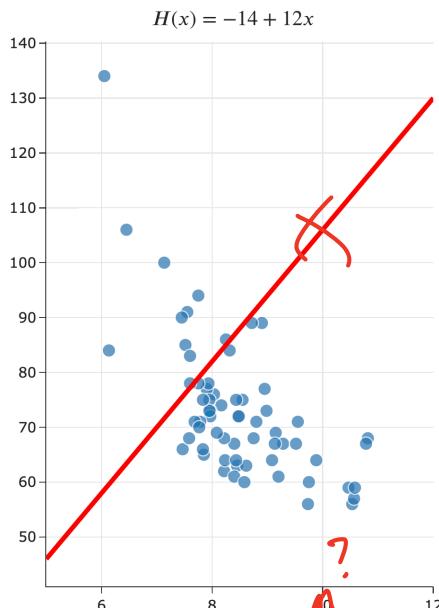


Hypothesis functions and parameters

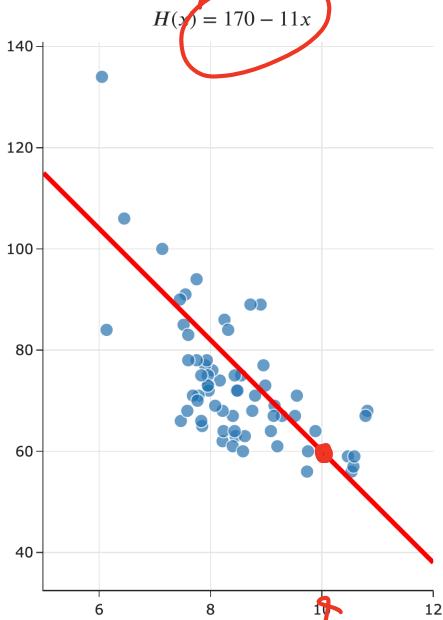
A hypothesis function, H , takes in an x as input and returns a predicted y .

Parameters define the relationship between the input and output of a hypothesis function.

The simple linear regression model, $H(x) = w_0 + w_1x$, has two parameters: w_0 and w_1 .



e.g.

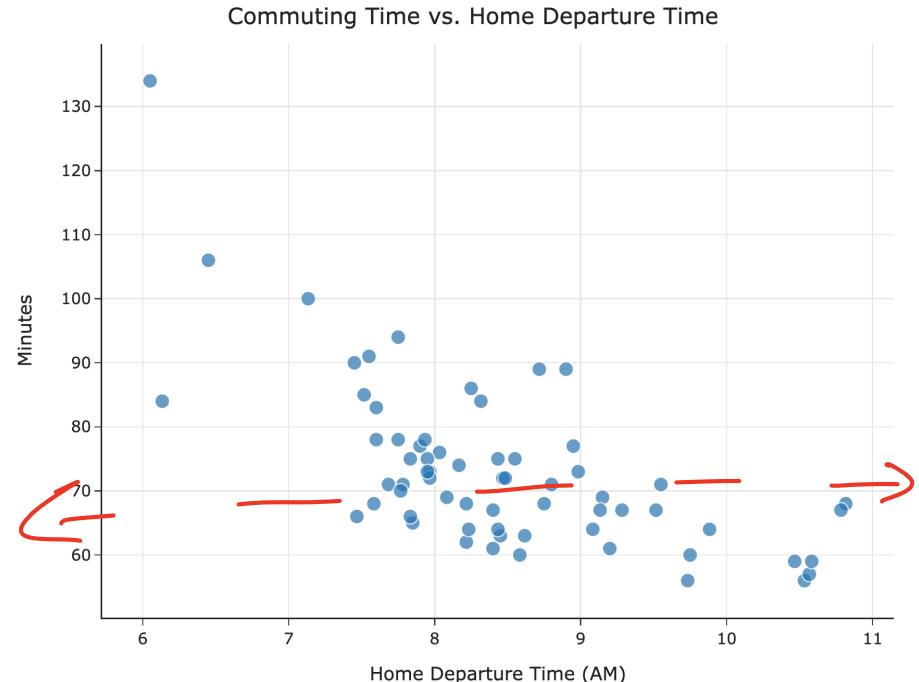


$$H(10) = 170 - 11(10) = 60 \text{ min @ 10 am}$$

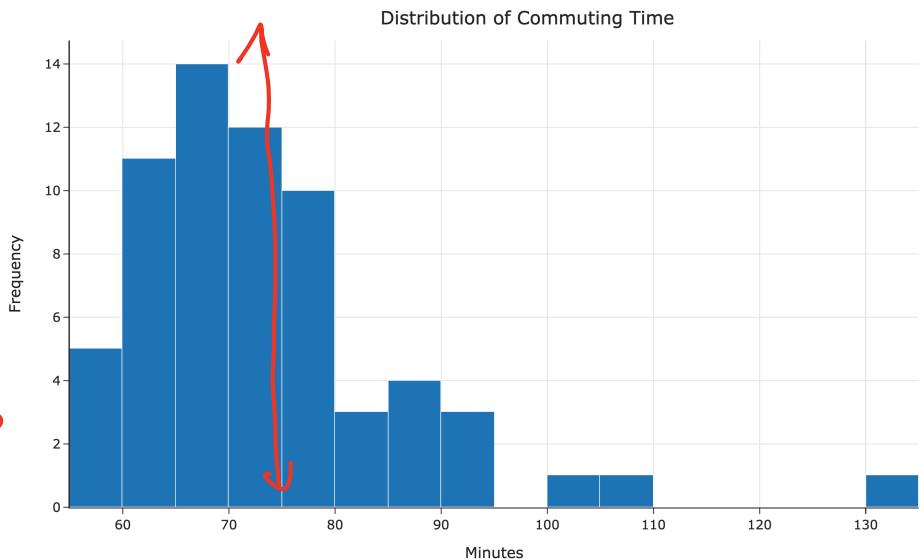
Changing parameter changes
how we make predictions!

The constant model

The constant model



how to draw a horiz. line to predict?



where do we draw a vertical line to summarize distribution

A concrete example

Let's suppose we have just a smaller dataset of just five historical commute times in minutes.

$$y_1 = 72$$

$$y_2 = 90$$

$$y_3 = 61$$

$$y_4 = 85$$

$$y_5 = 92$$

Given this data, can you come up with a prediction for your future commute time? How?

mean/average: 80

median: 65

eyeball a histogram

midrange: 76.5

↳ $\frac{\text{min} + \text{max}}{2}$

min/ max/ mode

most recent?

Some common approaches

- The mean:

$$\frac{1}{5}(72 + 90 + 61 + 85 + 92) = \boxed{80}$$

- The median:

61 72 **85** 90 92

- Both of these are familiar **summary statistics** – they summarize a collection of numbers with a single number.
- But which one is better? Is there a "best" prediction we can make?

The cost of making predictions

A loss function quantifies how bad a prediction is for a single data point.
↳ cost
↳ bad → we want loss to be low

- If our prediction is close to the actual value, we should have low loss.
- If our prediction is far from the actual value, we should have high loss.

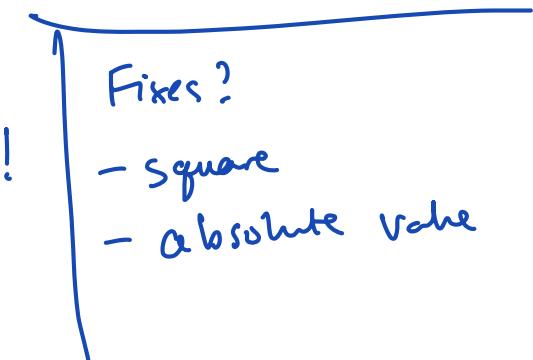
A good starting point is error, which is the difference between **actual** and **predicted** values.

$$e_i = \text{actual commute} - \text{predicted commute}$$
$$e_i = y_i - H(x_i)$$

Suppose my commute **actually** takes 80 minutes.

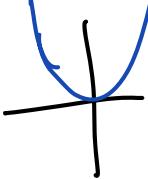
- If I predict 75 minutes: $\text{error} = 80 - 75 = 5$
- If I predict 72 minutes: $80 - 72 = 8$
- If I predict 100 minutes: $80 - 100 = -20$] low, but bad!

Issue: Some errors are negative
some errors are positive



Squared loss

One loss function is squared loss, L_{sq} , which computes $(\text{actual} - \text{predicted})^2$.

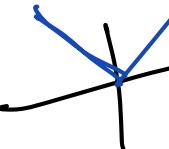
$$x^2 =$$


$$L_{\text{sq}}(y_i, H(x_i)) = (y_i - H(x_i))^2$$

\downarrow

$$w_0 + w_1 \cdot x$$

Note that for the constant model, $H(x_i) = h$, so we can simplify this to:

$$|x| =$$


$$L_{\text{sq}}(y_i, h) = (y_i - h)^2$$

\Downarrow

$$H(x_i)$$

$$\text{Note : } (y_i - h)^2 = (h - y_i)^2$$

Squared loss is not the only loss function that exists! Soon, we'll learn about absolute loss.

↳ main benefit: differentiable

A concrete example, revisited

Consider again our smaller dataset of just five historical commute times in minutes.

Suppose we predict the median, $h = 85$. What is the squared loss of 85 for each data point?

$$y_1 = 72 \rightarrow (72 - 85)^2 = 169$$

$$y_2 = 90 \rightarrow (90 - 85)^2 = 25$$

$$y_3 = 61 \rightarrow (61 - 85)^2 = 576$$

$$y_4 = 85 \rightarrow (85 - 85)^2 = 0$$

$$y_5 = 92 \rightarrow (92 - 85)^2 = 49$$

Goal : find a single number to summarize the loss of my prediction (85) across the whole dataset.

Averaging squared losses

We'd like a single number that describes the quality of our predictions across our entire dataset. One way to compute this is as the **average of the squared losses**.

- For the median, $h = 85$:

$$\frac{1}{5}((72 - 85)^2 + (90 - 85)^2 + (61 - 85)^2 + (85 - 85)^2 + (92 - 85)^2) = \boxed{163.8}$$

- For the mean, $h = 80$:

$$\frac{1}{5}((72 - 80)^2 + (90 - 80)^2 + (61 - 80)^2 + (85 - 80)^2 + (92 - 80)^2) = \boxed{138.8}$$

Low loss is good!

Which prediction is better? Could there be an even better prediction?

So: better it has lower average squared loss.

Mean squared error

more commonly used

- Another term for average squared loss is mean squared error (MSE).
- The mean squared error on our smaller dataset for any prediction h is of the form:

$$R_{\text{sq}}(h) = \frac{1}{5} ((72 - h)^2 + (90 - h)^2 + (61 - h)^2 + (85 - h)^2 + (92 - h)^2)$$

R stands for "risk", as in "empirical risk." We'll see this term again soon.

- For example, if we predict $h = 100$, then:

$$\begin{aligned} R_{\text{sq}}(100) &= \frac{1}{5} ((72 - 100)^2 + (90 - 100)^2 + (61 - 100)^2 + (85 - 100)^2 + (92 - 100)^2) \\ &= 538.8 \end{aligned}$$

- We can pick any h as a prediction, but the smaller $R_{\text{sq}}(h)$ is, the better h is!

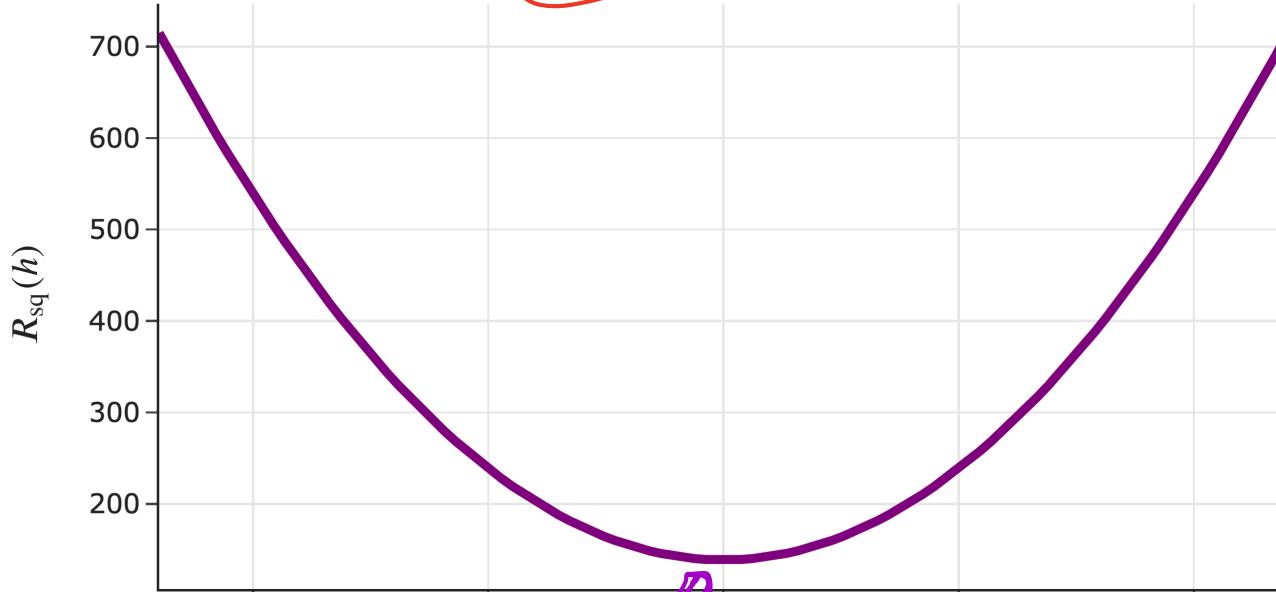
L: loss for a single data point

R: average loss over all data points

Visualizing mean squared error

parabola at vertex $h=90$

$$R_{\text{sq}}(h) = \frac{1}{5} ((72 - h)^2 + (90 - h)^2 + (61 - h)^2 + (85 - h)^2 + (92 - h)^2)$$



Which h corresponds to the vertex of $R_{\text{sq}}(h)$?

h to min MSE

Mean squared error, in general

- Suppose we collect n commute times, y_1, y_2, \dots, y_n .
- The mean squared error of the prediction h is:

$$h_{sq}(h) = \frac{1}{n} \left[(y_1 - h)^2 + (y_2 - h)^2 + \dots + (y_n - h)^2 \right]$$

- Or, using summation notation:

$$h_{sq}(h) = \frac{1}{n} \sum_{i=1}^n (y_i - h)^2$$

Exercise:

write some
code to do this

The best prediction

$$R_{\text{sq}}(h) = \frac{1}{n} \sum_{i=1}^n (y_i - h)^2$$

- We want the **best** prediction, h^* . Make $R_{\text{sq}}(h^*)$ as small as possible
- The smaller $R_{\text{sq}}(h)$ is, the better h is.
- **Goal:** Find the h that minimizes $R_{\text{sq}}(h)$.
The resulting h will be called h^* .
- **How do we find h^* ?**

↳ use Calculus!

Summary, next time

- We started with the abstract problem:
 - | Given historical commute times, predict your future commute time.
- We've turned it into a formal optimization problem:
 - | Find the prediction h^* that has the smallest mean squared error $R_{\text{sq}}(h)$ on the data.
- Implicitly, we introduced a three-step modeling process that we'll keep revisiting:
 - i. Choose a model.
 - ii. Choose a loss function.
 - iii. Minimize average loss, R .
- **Next time:** We'll solve this optimization problem by hand.