EECS 245, Winter 2026

LEC 16

Regression using Linear Algebra

→ Read Ch. 7.1, 7.2

Agenda

Chapter 7

Today's lecture answers:

"How does any of this material relate to linear regression and ML?"

- Brief recap: projecting onto colsp(X)
- Recap: Simple linear regression
- The design matrix
- Using multiple features

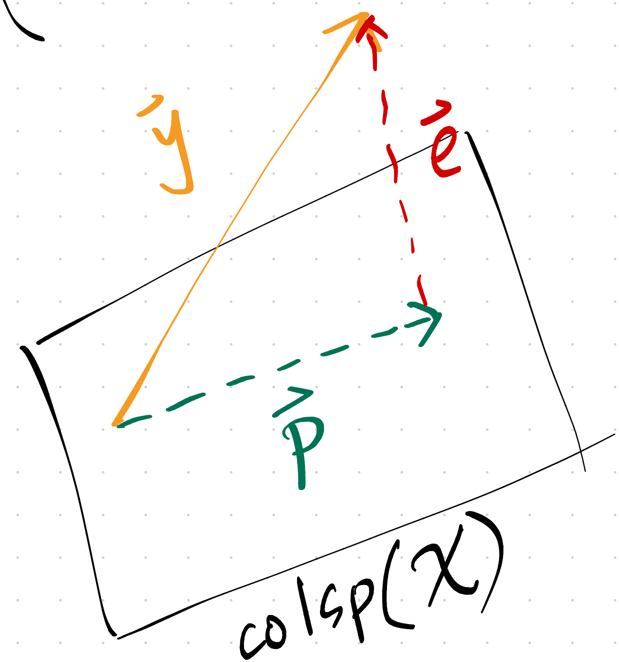
Announcements

- HW 8 due Saturday (one day extension)
- Lots of videos posted on course website
- Grade Report updated
- Midterm 2 is in 2 weeks...

  now is the time to lock in!

Recap: Projecting onto the column space

X n rows, d cols



$\vec{y} \in \mathbb{R}^n$

$\vec{p}$ is defined such that

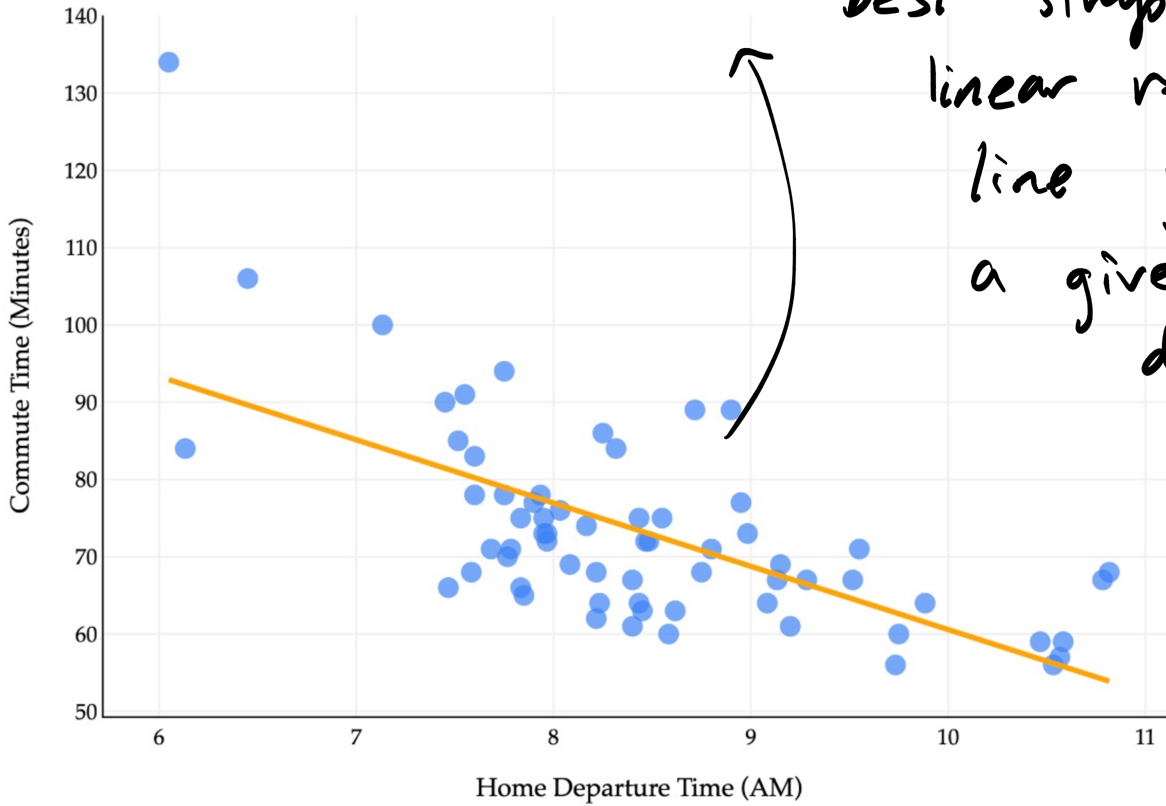
$\vec{e} = \vec{y} - \vec{p}$ is orthogonal to colsp(X)

$\vec{p} = X\vec{w}^*$

$\vec{w}^*$ satisfies $X^T X \vec{w} = X^T \vec{y}$

normal equation

Q: How did we find the "best" simple linear regression line for a given dataset?



Three step modeling recipe

① Choose a model

$$h(x_i) = w_0 + w_1 x_i$$

simple linear regression

② Choose a loss function

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

actual

pred

③ Minimize average loss →

$$R_{sq}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i) \right)^2$$

"empirical risk" "mean squared error"

Goal: Find w_0^* and w_1^*

without using partial derivatives, like we did pre-Midterm 1

$$R_{sq}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i) \right)^2$$

Given a dataset $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$

"Observation vector"

$$\vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

e.g. a vector of <u>actual</u> commute times

"Prediction vector"

$$\vec{p} = \begin{bmatrix} w_0 + w_1 x_1 \\ w_0 + w_1 x_2 \\ \vdots \\ w_0 + w_1 x_n \end{bmatrix}_{n \times 1} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}_{n \times 2} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}_{2 \times 1}$$

vector of predicted commute times

"design matrix"

"parameter vector"

$\vec{w}$

$$R_{sq}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i) \right)^2$$

$$\vec{e} = \vec{y} - \vec{p} = \vec{y} - X\vec{w} = \begin{bmatrix} y_1 - (w_0 + w_1 x_1) \\ y_2 - (w_0 + w_1 x_2) \\ \vdots \end{bmatrix}$$

error vector

$$\|\vec{e}\|^2 = \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i) \right)^2$$

$$R_{sq}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

We've converted

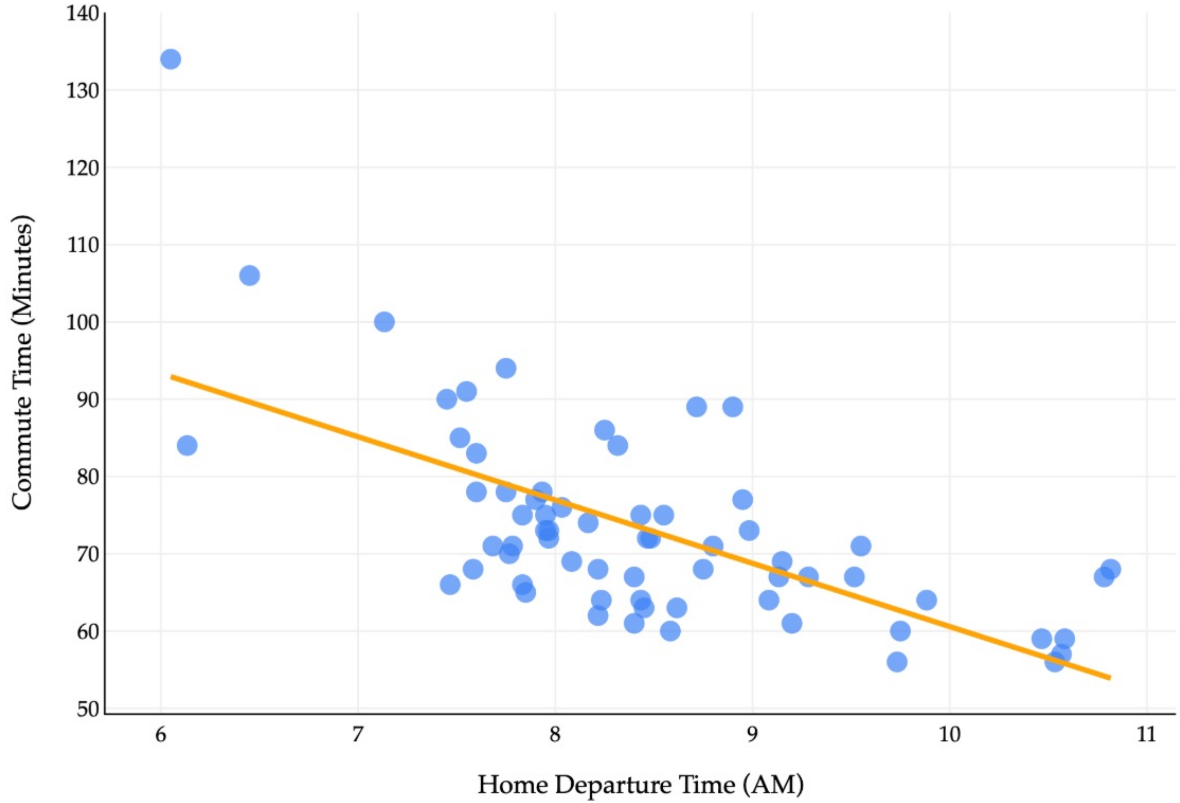
$$R_{sq}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i)\right)^2$$

into

$$R_{sq}(\vec{w}) = \frac{1}{n} \| \vec{y} - X\vec{w} \|^2$$

We already know — from last lecture — how to find the best $\vec{w}$!

It just needs to satisfy $X^T X \vec{w} = X^T \vec{y}$


140
130
120
110
100
90
80
70
60
50
Commute Time (Minutes)
6
7
8
9
10
11
Home Departure Time (AM)

$$X^T X \vec{w} = X^T \vec{y}$$

If $X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$, $\vec{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$, $\vec{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}$,

then

$$\vec{w}^* = (X^T X)^{-1} X^T \vec{y} = \begin{bmatrix} \bar{y} - w_1^* \bar{x} \\ \dfrac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \end{bmatrix}$$

optimal parameter vector

the same formulas from Chapter 2!

Why bother with this conversion?

⇒ now, we can add more features easily!

	departure_hour	day	day_of_month	minutes
0	10.816667	Mon	15	68.0
1	7.750000	Tue	16	94.0
2	8.450000	Mon	22	63.0
3	7.133333	Tue	23	100.0
4	9.150000	Tue	30	69.0

$\vec{y}$

the only input variable so far

can we use this too?

$$h(dh_i, dom_i) = w_0 + w_1 dh_i + w_2 dom_i$$

departure hour$_i$

day of month$_i$

we used to call this x_i

multiple features: multiple linear regression

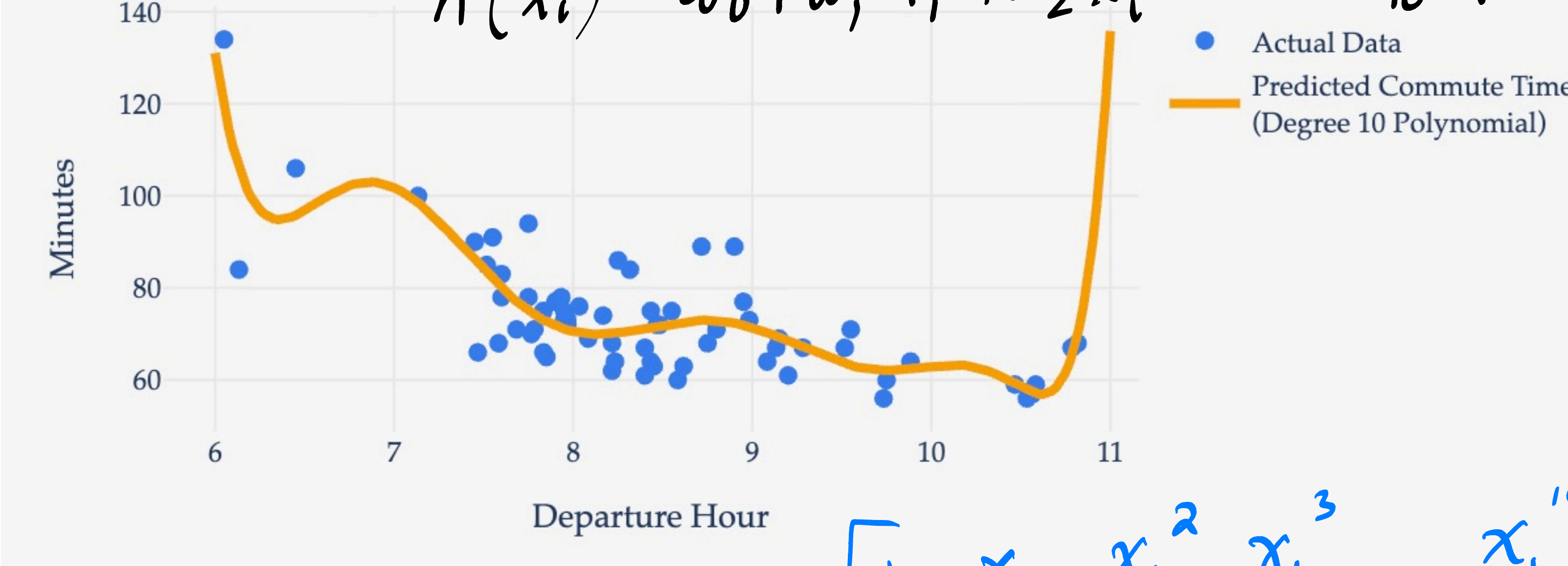
2 features → 3 parameters

	departure_hour	day	day_of_month	minutes
0	10.816667	Mon	15	68.0
1	7.750000	Tue	16	94.0
2	8.450000	Mon	22	63.0
3	7.133333	Tue	23	100.0
4	9.150000	Tue	30	69.0

First 2 rows of this design matrix?

$$X_{n \times 3} = \begin{bmatrix} 1 & 10.816\ldots & 15 \\ 1 & 7.75 & 16 \\ & \vdots & \end{bmatrix}$$

$$h(x_i) = w_0 + w_1 x_i + w_2 x_i^2 + \dots + w_{10} x_i^{10}$$



	departure_hour	day	day_of_month	minutes
0	10.816667	Mon	15	68.0
1	7.750000	Tue	16	94.0
2	8.450000	Mon	22	63.0
3	7.133333	Tue	23	100.0
4	9.150000	Tue	30	69.0

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & \cdots & x_1^{10} \\ 1 & x_2 & x_2^2 & x_2^3 & & x_2^{10} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \end{bmatrix}$$

Any model that is a linear combination of features (weights are w's)

can be fit using linear regression,

i.e. where $\vec{w}^*$ satisfies $X^T X \vec{w} = X^T \vec{y}$

$$h(dh_i, dom_i) = w_0 \cdot 1 + w_1\, dh_i + w_2 \tan\left(\frac{dh_i^2}{\cos(dom_i)}\right)$$

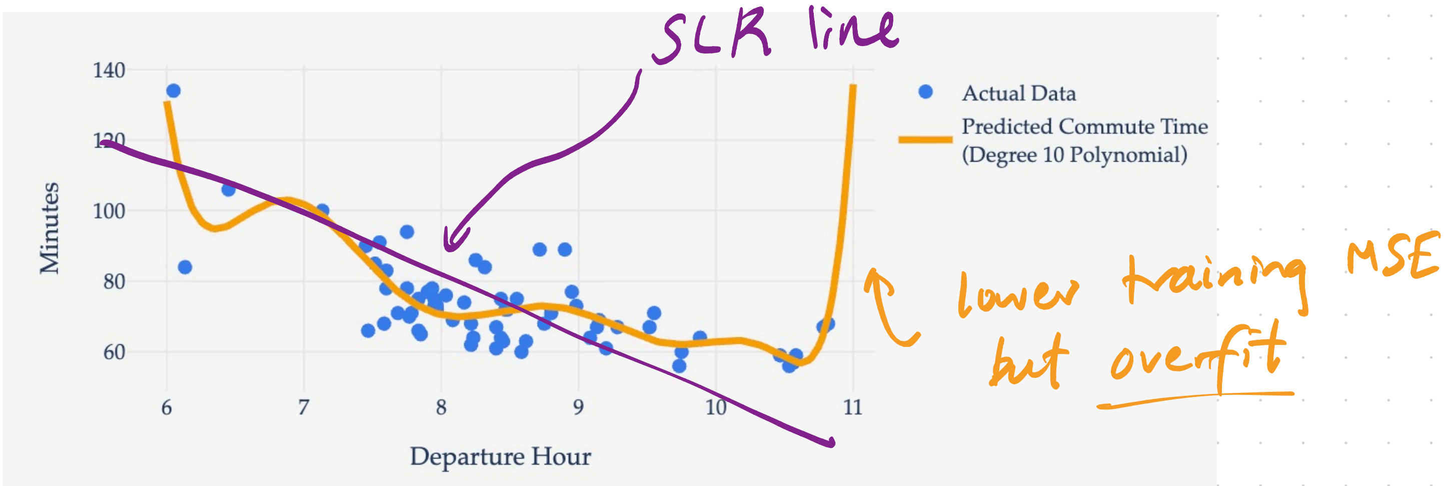
good: "linear in the parameters"

$$h(dh_i, dom_i) = w_0 + \sin(w_1\, dh_i)$$

can't use linear regression!

Adding new features can't hurt Mean Squared Error

on the training data



but could have worse performance on unseen test data which is ultimately what matters issue : OHE introduces linear dependency (with 1's column)

	day	day == Mon	day == Tue	day == Wed	day == Thu	day == Fri
1	Mon	$\boxed{1}$	0	0	0	0
1	Tue	0	$\boxed{1}$	0	0	0
	Mon	$\boxed{1}$	0	0	0	0
⋮	...	...	...	...	...	...
	Mon	$\boxed{1}$	0	0	0	0
	Tue	0	$\boxed{1}$	0	0	0
1	Thu	0	0	0	$\boxed{1}$	0

one hot encoding → how we add categorical features