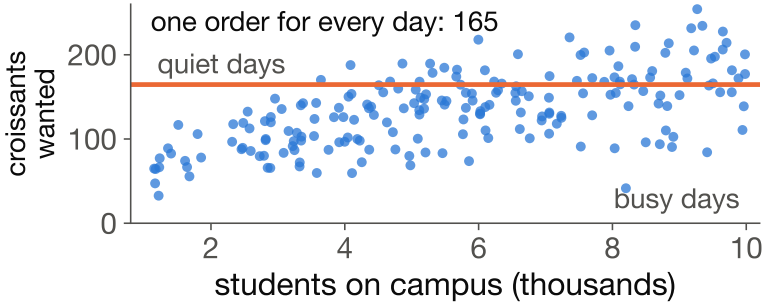
Decisions that use features

208424 Lecture 13

Lecture 12: one order for every day

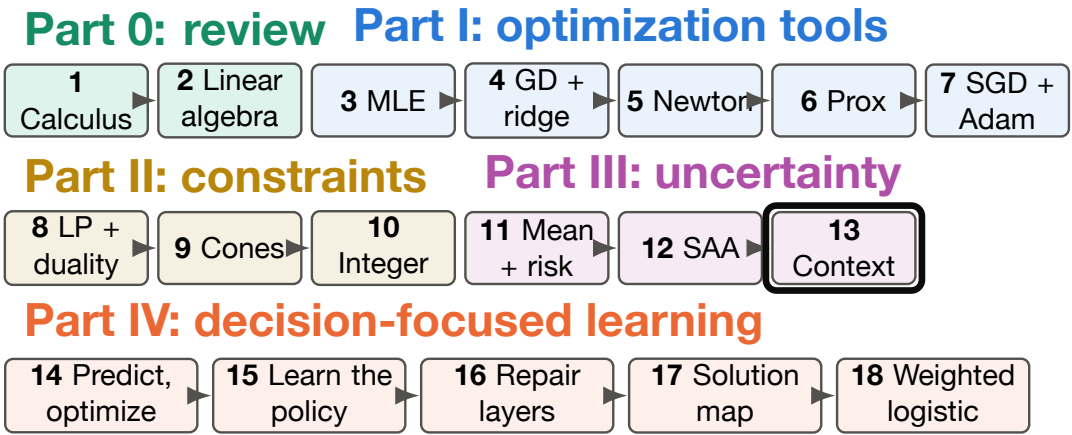


In Lecture 12, every day got the same order.

On busy days, the demand is much higher.

Today, busy days get a larger order.

Course map


Part 0: review
Part I: optimization tools
1 Calculus
2 Linear algebra
3 MLE
4 GD + ridge
5 Newton
6 Prox
7 SGD + Adam
Part II: constraints
Part III: uncertainty
8 LP + duality
9 Cones
10 Integer
11 Mean + risk
12 SAA
13 Context
Part IV: decision-focused learning
14 Predict, optimize
15 Learn the policy
16 Repair layers
17 Solution map
18 Weighted logistic

Lecture 13: Decisions that use features

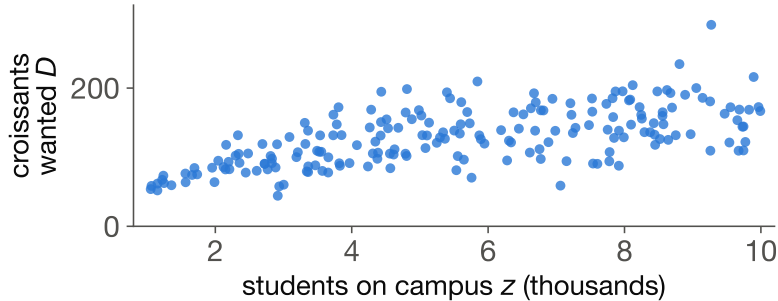


Photo: Athikhun.suw, CC BY 4.0, Wikimedia Commons

A café knows tomorrow's number of students.

A hospital knows the day and the time of day.

The café's past days

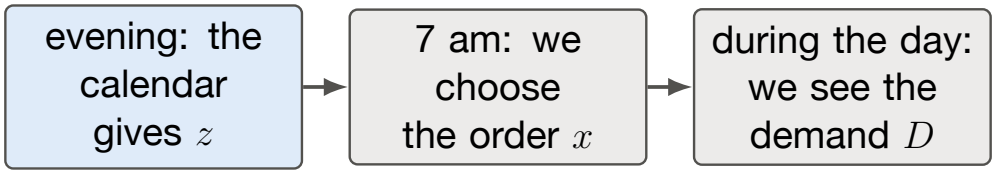


Each dot is one past day of the café.

The number of students on campus is z.

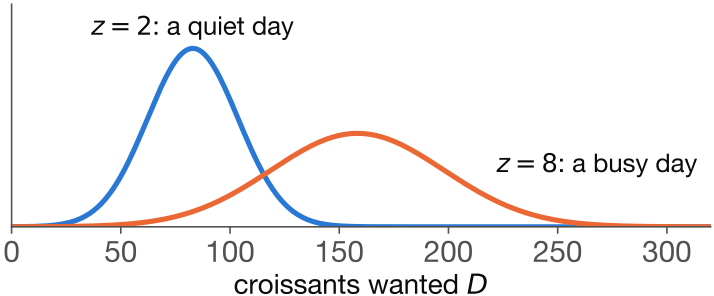
On average, the demand grows with z.

A feature



We know tomorrow's z before we order.
A number that we know before we decide is called a feature.

Demand on a quiet day and a busy day

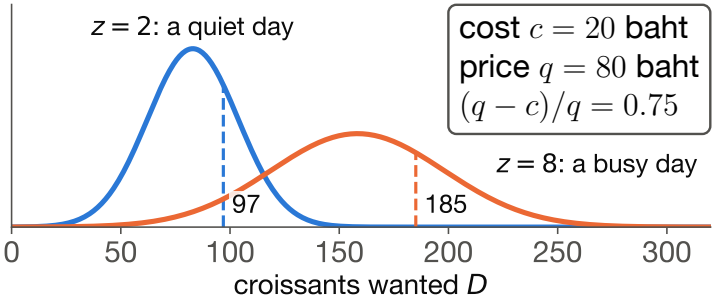


With $z = 2$, the average demand is 83.

With $z = 8$, the average demand is 158.

Busy days also have a larger SD.

The optimal order for $z = 2$ and $z = 8$

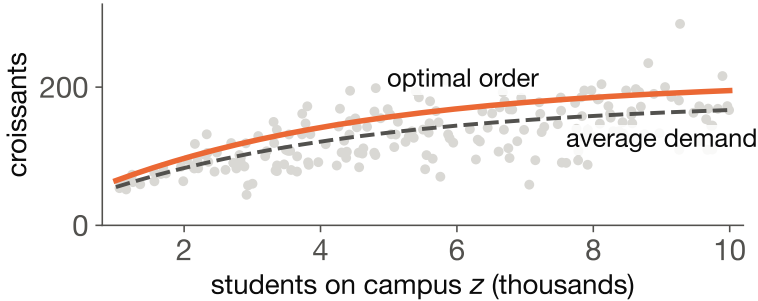


As in Lecture 11, we order the 0.75 quantile.

For $z = 2$, the optimal order is 97.

For $z = 8$, the optimal order is 185.

The optimal order for every z

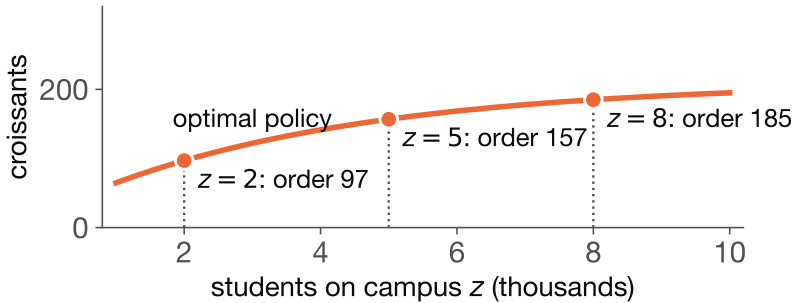


The dashed curve shows the average demand.

The solid curve shows the optimal order.

The optimal order is above the average.

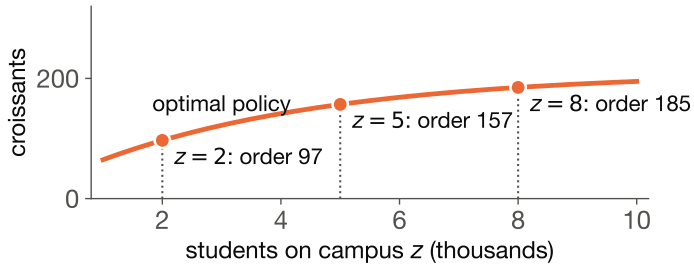
A policy



A policy gives an order $x(z)$ for each z.

The solid curve is the optimal policy.

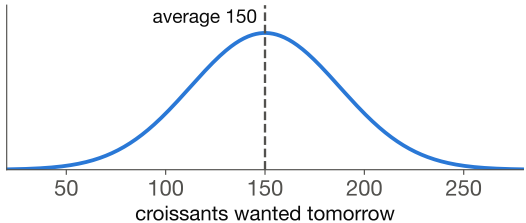
Summary: a feature



The optimal order depends on the feature z.

A policy gives an order $x(z)$ for each z.

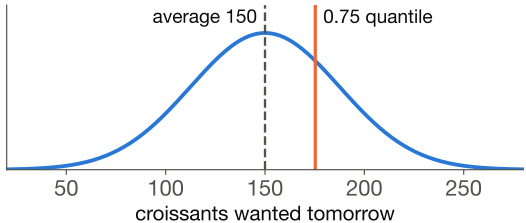
Checkpoint 1



Tomorrow's demand has an average of 150.

Should the café order 150 croissants?

Checkpoint 1



Tomorrow's demand has an average of 150.

Should the café order 150 croissants?

No. The optimal order is the 0.75 quantile.

The 0.75 quantile of this demand is above 150.

Four methods to use z

1. ignore z:
SAA (Lecture 12)

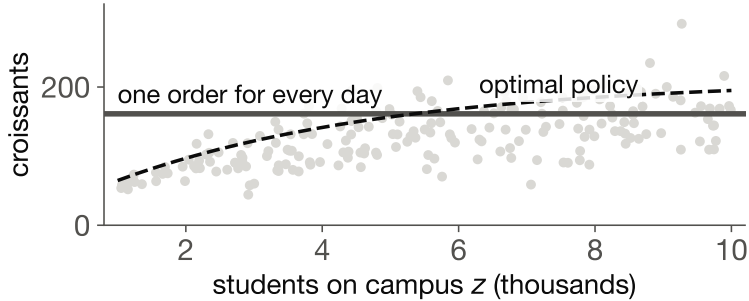
2. predict,
then optimize

3. local SAA

4. learn the
policy directly

Each method gives a policy from the past days.

Method 1: ignore z (Lecture 12)

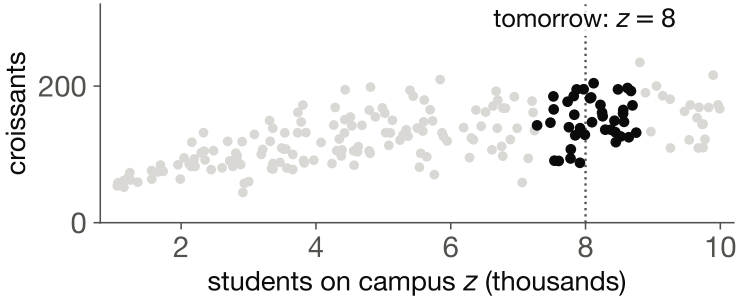


SAA uses one order for every day.

On quiet days, this order is too large.

On busy days, this order is too small.

The k nearest days

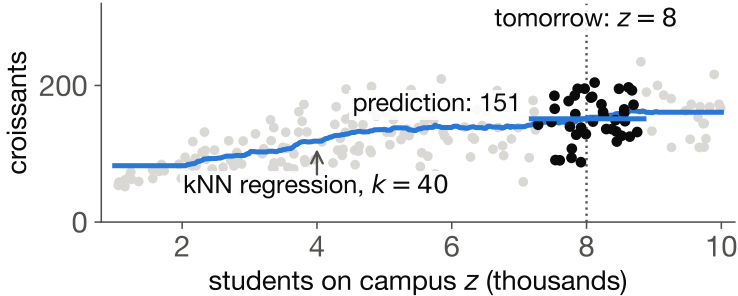


Tomorrow, there are 8 thousand students.

We take the $k = 40$ past days with z closest to 8.

These days are the k nearest neighbors (kNN).

kNN regression

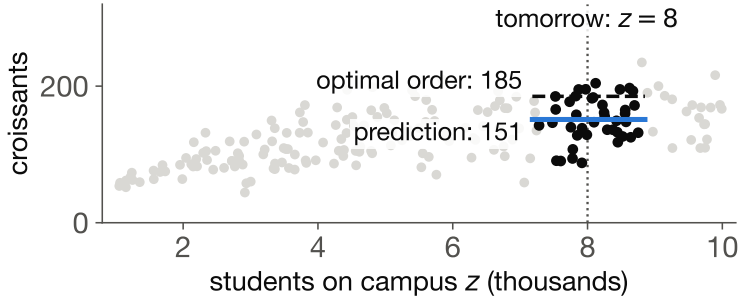


kNN regression predicts D from the k nearest days.

The prediction is the average demand of the k days.

kNN regression gives a prediction for every z.

Method 2: predict, then optimize

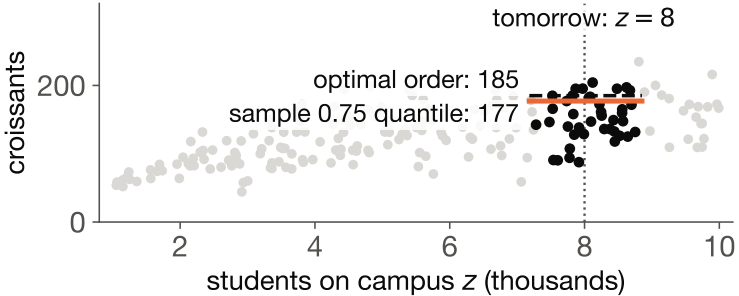


First, we predict D with kNN regression.

The predicted average for $z = 8$ is 151.

Then we order 151, as if D were exactly 151.

Method 3: local SAA

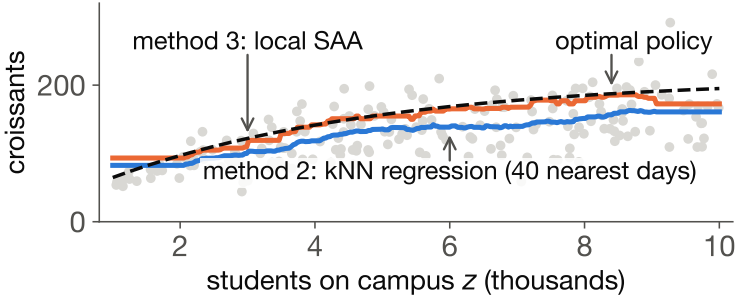


Local SAA uses SAA on the 40 nearest days.

The sample 0.75 quantile of these days is 177.

So local SAA orders 177 croissants.

Methods 2 and 3 for every z



For every z, we use the 40 nearest days.

Method 2 orders too few for almost every z.

Method 3 is close to the optimal policy.

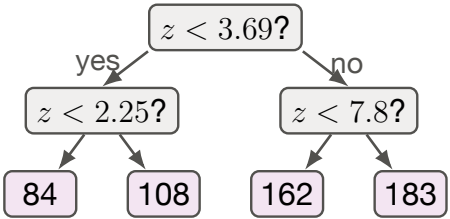
Local SAA with weights

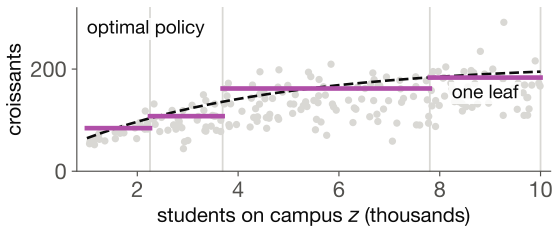
$$\text{maximize} \quad \sum_i \underbrace{w_i(z)}_{\text{weight of past day } i} \underbrace{\left(80 \min(D_i, x) - 20x\right)}_{\text{profit on day } i}$$

Each of the k nearest days gets weight $1/k$.

All other days get weight 0.

A regression tree





A regression tree splits the past days by z.

Each group of days is called a leaf.

In each leaf, we order the sample 0.75 quantile.

Training days and validation days

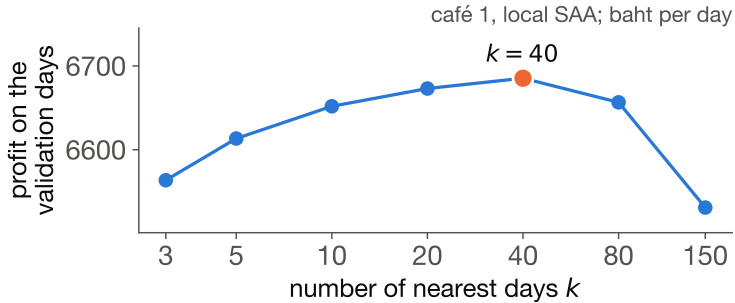


The 200 days in the pictures are training days.

The café also has 100 validation days.

We choose k on the validation days (Lecture 4).

Choosing k

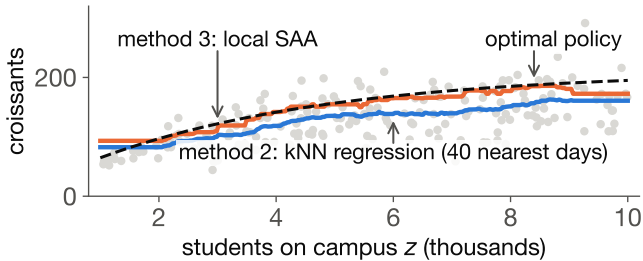


Each k gives a profit on the validation days.

This profit is highest at $k = 40$.

So we choose $k = 40$.

Summary: the k nearest days



Methods 2 and 3 both use the k nearest days.

Method 2 orders the predicted average.

Method 3 orders the sample 0.75 quantile.

Checkpoint 2

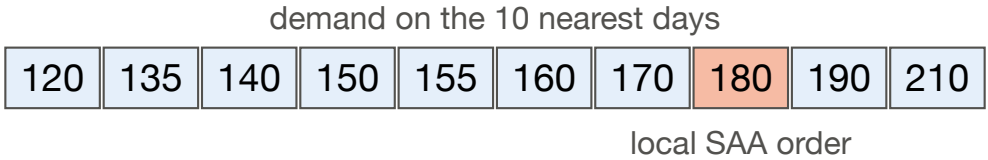
demand on the 10 nearest days

120	135	140	150	155	160	170	180	190	210

The cost is 20 baht, and the price is 80 baht.

What does local SAA order for tomorrow?

Checkpoint 2



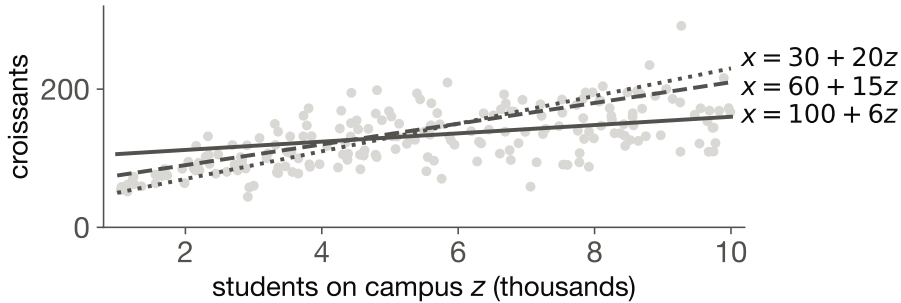
The cost is 20 baht, and the price is 80 baht.

What does local SAA order for tomorrow?

Since $0.75 \times 10 = 7.5$, we take the 8th value.

So local SAA orders 180 croissants.

Method 4: a linear policy

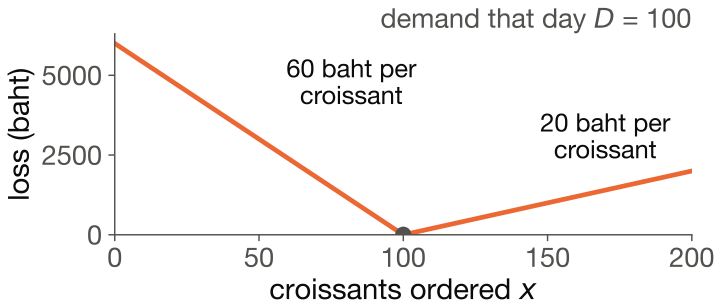


A linear policy has the form $x(z) = b_0 + b_1 z$.

Each line in the picture is one policy.

Method 4 chooses b_0 and b_1 without predicting D.

The pinball loss (Lecture 11)



Each missing croissant adds 60 baht to the loss.

Each leftover croissant adds 20 baht to the loss.

The average pinball loss

On past day i, the policy orders $b_0 + b_1 z_i$.

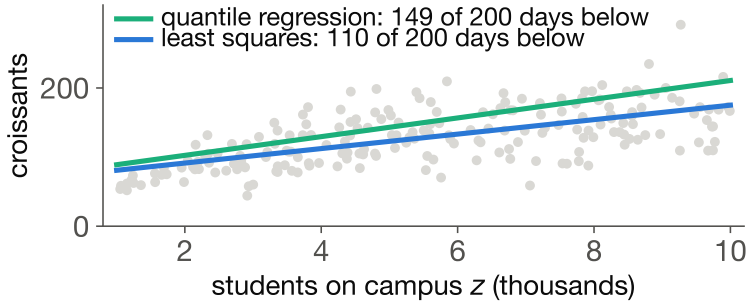
pinball loss (Lecture 11)

$$\text{minimize} \underbrace{\frac{1}{n} \sum_{i=1}^{n}} \ell\left(b_0 + b_1 z_i,\ D_i\right)$$

average over
the n training days

We choose b_0 and b_1 to minimize this average.

Quantile regression

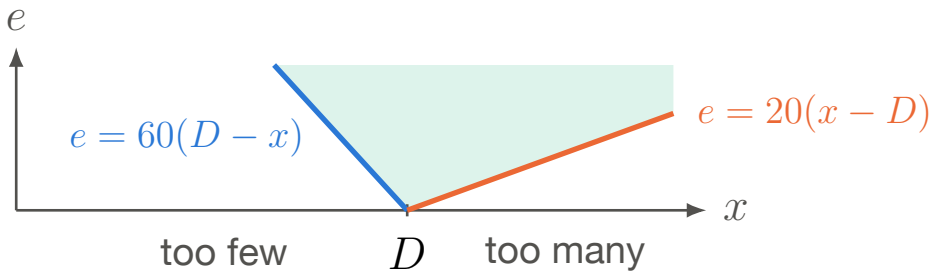


Least squares fits the average (Lecture 3).

Quantile regression fits the 0.75 quantile.

Quantile regression uses the pinball loss.

The pinball loss as two constraints



$$e \geq 60(D - x) \text{ and } e \geq 20(x - D)$$

The pinball loss is the smallest e that satisfies both.

We will find b_0 and b_1 with an LP, as in Lecture 8.

The pinball loss on each day

We add a variable e_i for the loss on day i.

$$e_i \geq 60(D_i - x_i) \quad \text{and} \quad e_i \geq 20(x_i - D_i)$$

$x_i = b_0 + b_1 z_i$, the order on day i

The decision variables are b_0, b_1 and $e_1, \ldots, e_n$.

Quantile regression as an LP

$$\text{minimize} \quad \frac{1}{n}\sum_{i=1}^{n} e_i$$

subject to $e_i \geq 60(D_i - x_i),\ e_i \geq 20(x_i - D_i)$

The objective and the constraints are linear.

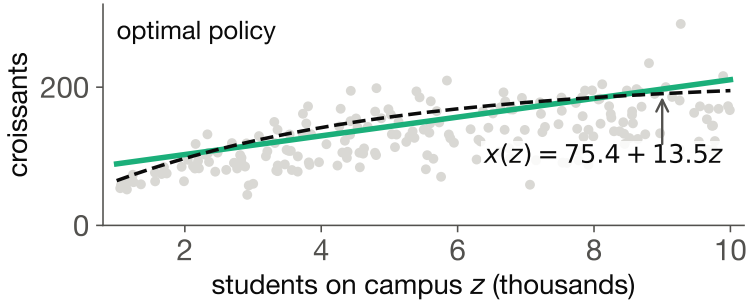
Quantile regression in Python

```
import cvxpy as cp
b0, b1 = cp.Variable(), cp.Variable()
e = cp.Variable(200)  # loss on each day
x = b0 + b1 * z       # order on each day
cons = [e >= 60 * (D - x), e >= 20 * (x - D)]
obj = cp.Minimize(cp.sum(e) / 200)
cp.Problem(obj, cons).solve()
b0.value, b1.value    # about 75.4 and 13.5
```

The arrays z and D hold the 200 training days.

cvxpy gives $b_0 = 75.4$ and $b_1 = 13.5$.

Method 4 for every z

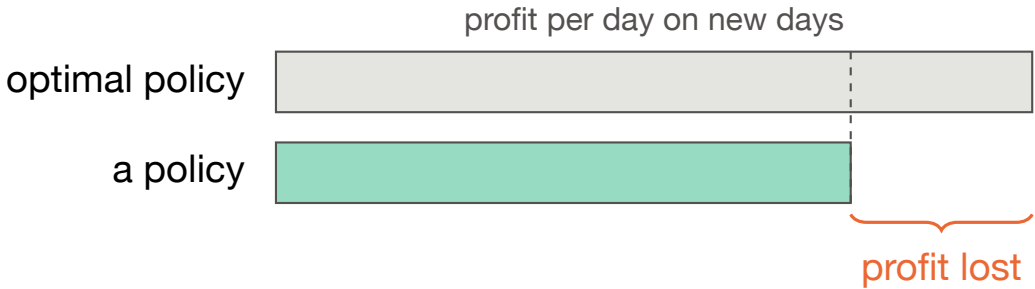


The fitted policy is $x(z) = 75.4 + 13.5z$.

The line is close to the optimal policy.

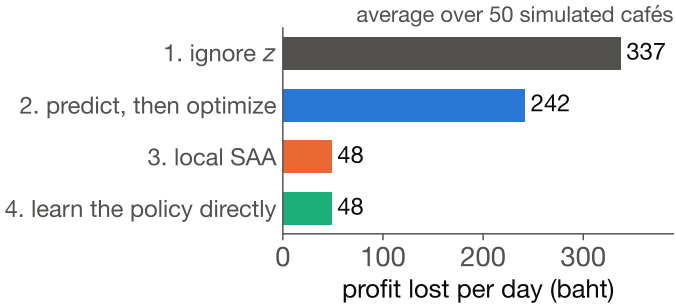
The optimal policy is not linear in z.

Profit lost



We evaluate each policy on new days.
The profit lost is the optimal policy's profit minus the policy's profit.

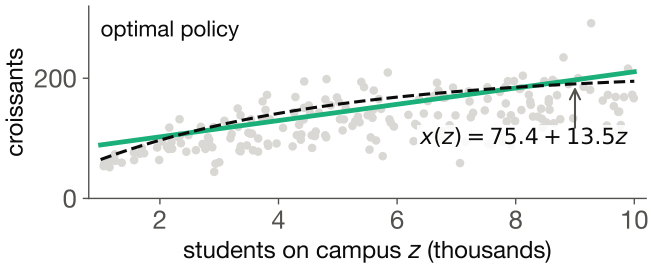
Which method works best?



Methods 3 and 4 lose about 48 baht per day.

Ordering the predicted average loses 242 baht.

Summary: learn the policy directly



Quantile regression fits a linear policy.

We solve an LP to find b_0 and b_1.

Checkpoint 3

a past day:
$z = 6, \;\; D = 170$

each missing croissant: adds 60 baht
each leftover croissant: adds 20 baht

The policy is $x(z) = 60 + 15z$.

What is the pinball loss on this day?

Checkpoint 3

a past day:
$z = 6,\ \ D = 170$

each missing croissant: adds 60 baht
each leftover croissant: adds 20 baht

The policy is $x(z) = 60 + 15z$.

What is the pinball loss on this day?

The order is $60 + 15 \times 6 = 150$ croissants.

The loss is $60 \times (170 - 150) = 1200$ baht.

More features for each day

f_1	f_2	f_3	$\cdots$	f_{20}	D
0.04	-0.89	-0.06	$\cdots$	-0.25	180
1.56	-0.58	-1.60	$\cdots$	1.26	172
-1.23	-0.18	0.28	$\cdots$	-0.54	102

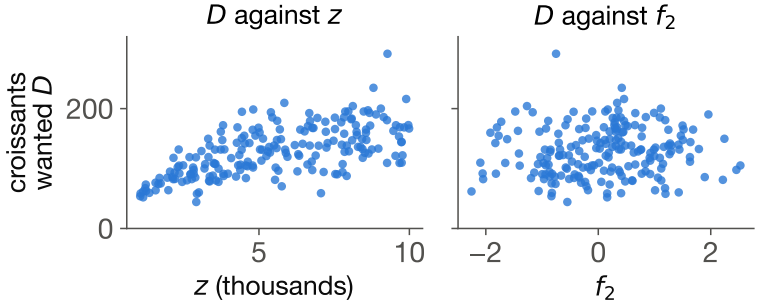
$$f_1 = (z - 5.5)/2.6$$

The café records 19 more features each day.

Each day now has $p = 20$ features.

We try $p = 1$, 2, 5, 10 and 20.

Features that do not affect demand

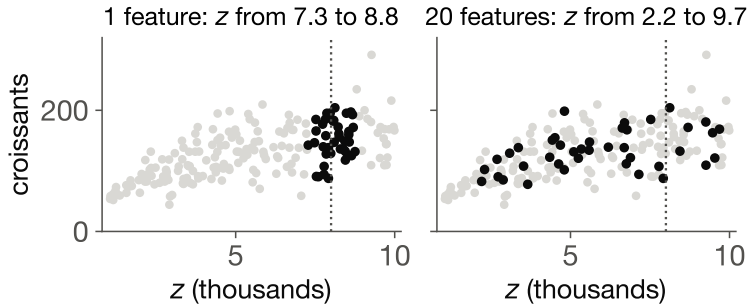


We simulated the 19 extra features at random.

These features do not affect demand.

Local SAA and the linear policy use all 20 features.

Nearest days with 20 features



The nearest days are found with all 20 features.

So the 40 nearest days are far apart in z.

A linear policy with many features

Each feature gets its own coefficient.

$$x = b_0 + b_1 \overbrace{f_1}^{(z - 5.5)/2.6} + b_2 f_2 + \cdots + b_p f_p$$

b_1: coefficient

With $p = 20$, we fit 21 coefficients.

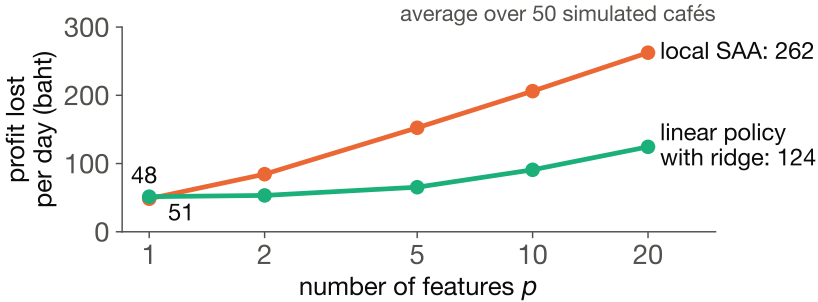
A ridge penalty (Lecture 4)

Ridge adds a penalty on the coefficients.

$$\text{minimize} \quad \frac{1}{n}\sum_{i=1}^{n} \ell(\underbrace{x_i}_{\text{the policy's order on day } i}, D_i) + \underbrace{\lambda\left(b_1^2 + \cdots + b_p^2\right)}_{\text{ridge penalty}}$$

We choose λ on the validation days.

Profit lost as features are added

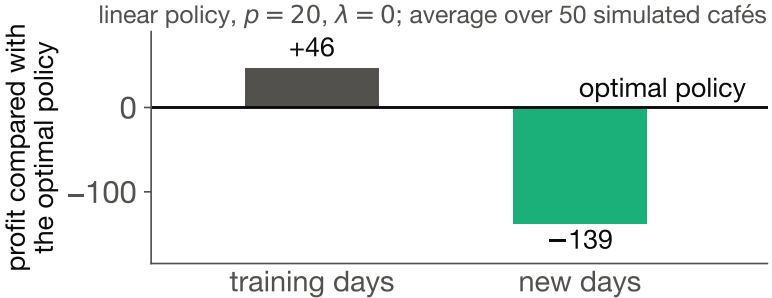


Both methods lose more profit as p grows.

At $p = 20$, local SAA loses 262 baht per day.

The linear policy with ridge loses 124.

Training days and new days

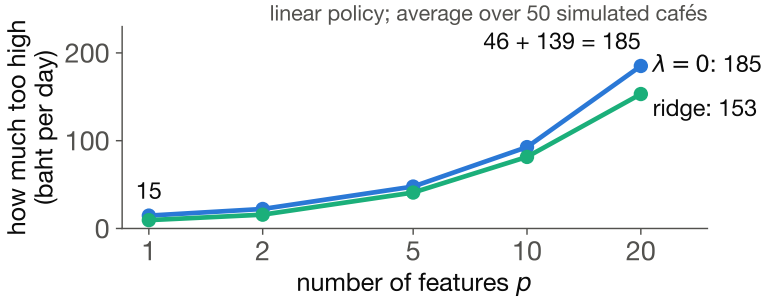


We compare the policy with the optimal policy.

On training days, the policy earns 46 baht more.

On new days, the policy earns 139 baht less.

Profit on training days is too high

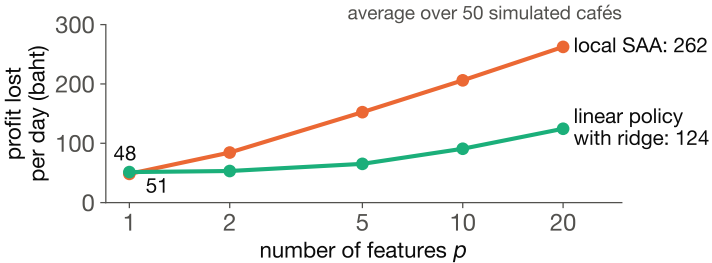


With $p = 20$, the profit is 185 baht too high.

With $p = 1$, the profit is only 15 baht too high.

Ridge lowers the 185 baht to 153 baht.

Summary: many features



Features unrelated to demand lower the profit.

Profit on training days is too high on average.

Checkpoint 4

the policy ($p = 20$, $\lambda = 0$):
7062 baht per day
on its 200 training days

the optimal policy:
6998 baht per day
on average

Will the policy earn about 7062 on new days?

Checkpoint 4

the policy ($p = 20$, $\lambda = 0$):
7062 baht per day
on its 200 training days

the optimal policy:
6998 baht per day
on average

Will the policy earn about 7062 on new days?

No. No policy earns more than 6998 on average.

Profit on training days is too high on average.

Nurses in an emergency room



Photo: Government of Thailand, CC BY 2.0, Wikimedia Commons

A hospital plans its nurses in advance.

Missing nurses are replaced by agency nurses.

Agency nurses cost more than regular nurses.

The same problem as the café

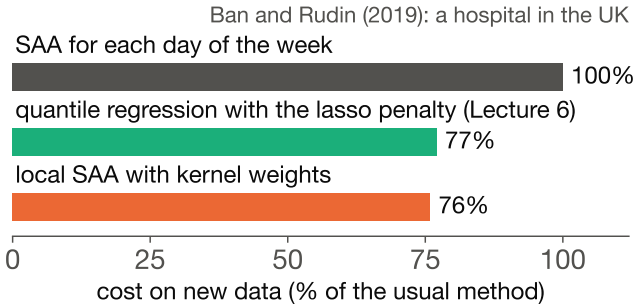
	café	hospital
decision	order x croissants	plan x nurses
demand	croissants wanted	nurses needed (1 nurse for 5 patients)
one too few	loses 60 baht	costs 2.5
one too many	loses 20 baht	costs 1
features	z	day, time, past demand

Planning nurses is a newsvendor problem.

The costs are in regular nurse wages.

The café's quantile is $60/(60+20) = 0.75$.

Features lower the cost

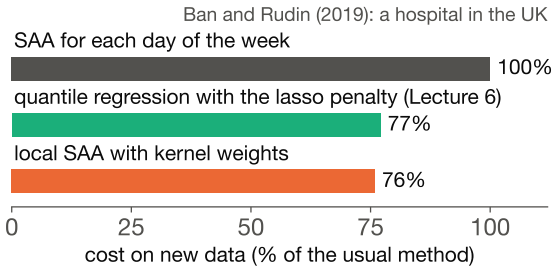


The usual method is SAA for each day of the week.

More features lowered the cost by 23% and 24%.

The savings are £44,219 and £46,555 per year.

Summary: nurse staffing



Planning nurses is a newsvendor problem.

Features related to demand lowered the cost.

Checkpoint 5

one nurse too few:
costs 2.5

one nurse too many:
costs 1

Which quantile should the hospital plan for?

Checkpoint 5

one nurse too few:
costs 2.5

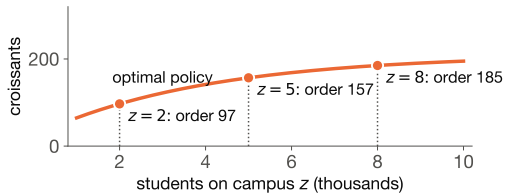
one nurse too many:
costs 1

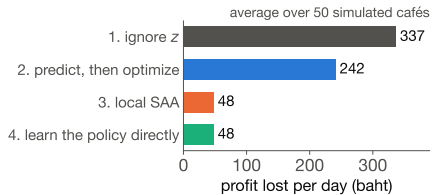
Which quantile should the hospital plan for?

The quantile is $2.5/(2.5+1) = 0.71$.

The plan has enough nurses 71% of the time.

Lecture summary



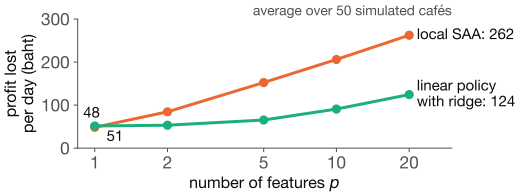


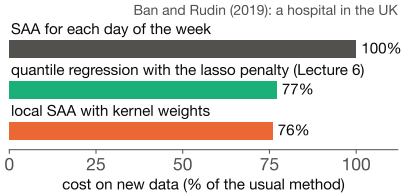
A policy gives an order for each value of z.

Ordering the predicted average loses profit.

Local SAA and quantile regression do much better.

Lecture summary (continued)



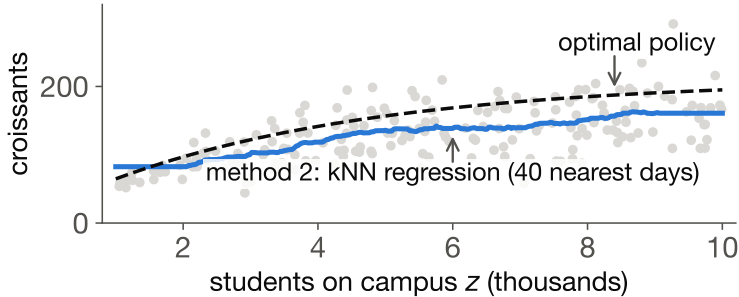


Features unrelated to demand lower the profit.

Profit on training days is too high on average.

Features related to demand lower the cost.

Next lecture

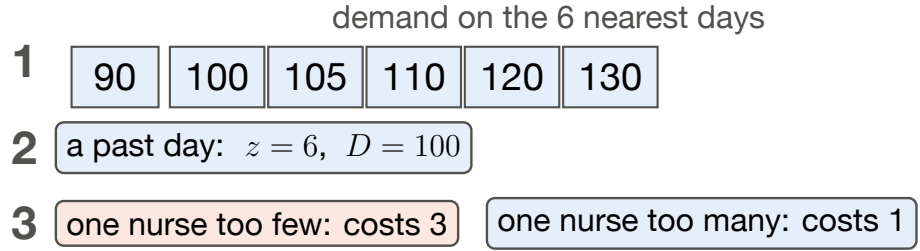


Today, method 2 lost 242 baht per day.

In Lecture 14, we study predict, then optimize.

When does a good prediction give a good decision?

Exercises



1. Find the local SAA order for $c = 20$ and $q = 80$.
2. Find the pinball loss of $x(z) = 30 + 15z$.
3. Find the quantile for these nurse costs.