

Algorithms with Calibrated Machine Learning Predictions

Ellen Vitercik, Stanford

ICML'25



Judy Hanwen Shen



Anders Wikum

ML for combinatorial optimization

Combinatorial opt has long captured ML researchers' imaginations

- E.g., traveling salesman problem (TSP): iconic for theory and ML
 - 40 years from Hopfield networks [e.g., Hopfield, Tank, '86] to LLMs

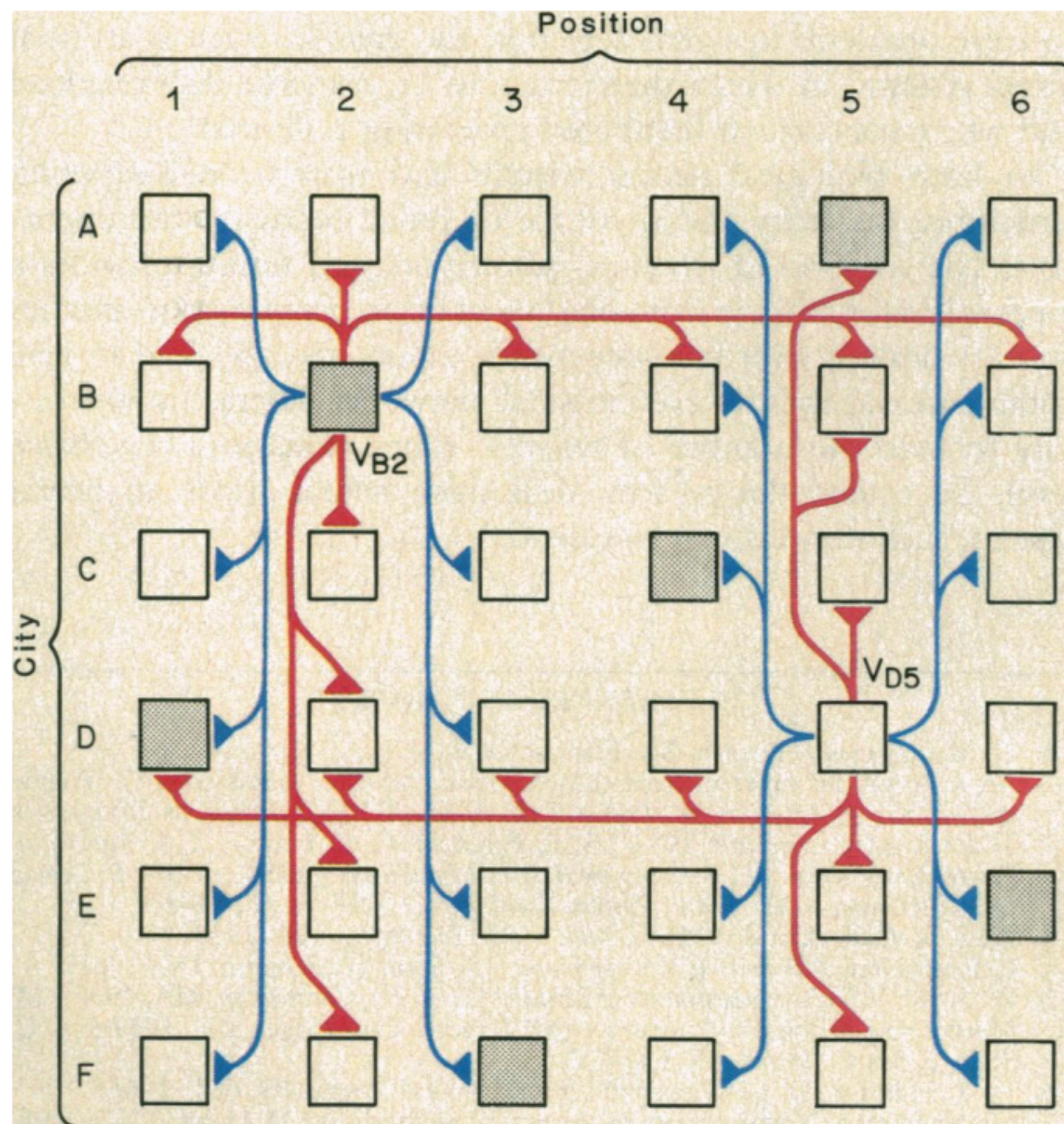


Fig. 5. A stylized picture of the syntax and connections of the **TSP neural circuit**. Each neuron is symbolically indicated by a square. The neurons are arranged in an n by n array. Each city is associated with n neurons in a row, and each position in the final tour is associated with n neurons in a column. A given neuron ($V_{X,j}$) represents the hypothesis that city X is in position j in the solution. The patterns of synaptic connection for two different neurons are indicated.

8 AUGUST 1986

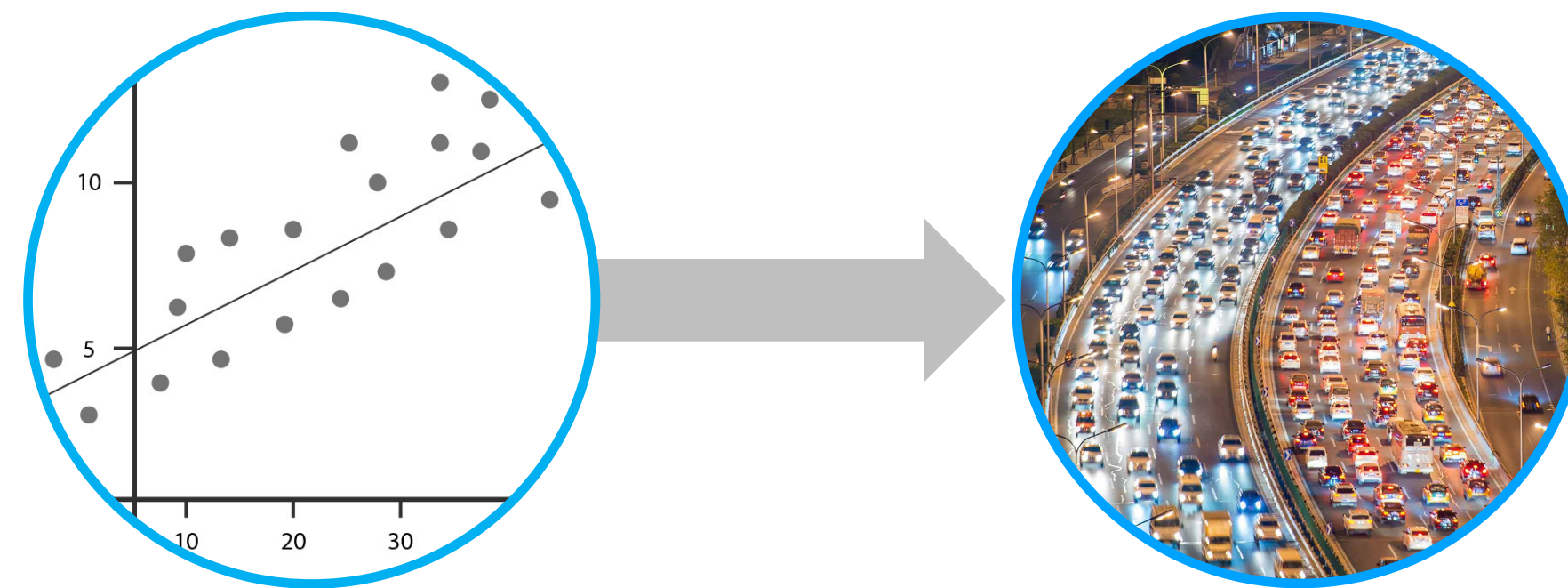
ML for combinatorial optimization

Combinatorial opt has long captured ML researchers' imaginations

- E.g., traveling salesman problem (TSP): iconic for theory and ML
 - 40 years from Hopfield networks [e.g., Hopfield, Tank, '86] to LLMs
- Hard problems recur on **structured instance distributions**
 - Data reveals patterns not leveraged by worst-case analysis
- ML can **guide decisions** at multiple interfaces
 - Policies, solver controls, heuristics, configurations
- Inputs are often **uncertain** or **partially observed**
 - Predictions help, but errors can amplify: don't blindly trust predictions

Algorithms and prediction uncertainty

- Inputs are often **uncertain** or **partially observed**
 - Predictions help, but errors can amplify: don't blindly trust predictions



- **Our jumping off point:** ML models can estimate uncertainty **automatically**
 - Examples: calibration and conformal predictions
 - Well-defined, statistical notion of whether a prediction can be trusted

Our contributions

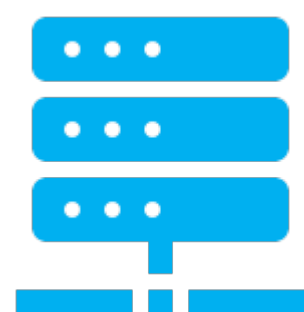
Demonstrate calibration's utility for online algorithms

Two case-studies:



Online rent-or-buy problems

- E.g., cloud vs equipment purchase, subscription vs lifetime licenses, ...
- Algorithm with guarantees that improve with accuracy and calibration error



Online job scheduling

- E.g., processing radiology diagnostic images
- Calibrated predictions yield better schedules than prior work [Cho et al., '22]

Validate methods on real-world datasets

Related work

Decision-making under uncertainty + machine learning + theory

Data-driven algorithm design

[e.g., Gupta, Roughgarden '16; Balcan, Nagarajan, [V](#), White, '17; Kleinberg et al., '17; Balcan, Dick, [V](#), '18; Garg, Kalai, '18; ...]

Algorithms with predictions

[e.g., book chapter by Mitzenmacher, Vassilvitskii, '20]

This talk: most in this style



Calibration for decision-making

[e.g., Foster, Vohra, GEB'97; Zhao et al., NeurIPS'21; Kleinberg et al., COLT'23; Roth, Shi, EC'23; Noarov et al., ICML'25; ...]

Data-driven robust optimization

[e.g., Esfahani, Kuhn, '17; Bertsimas et al., '18; ...]

Decision-focused learning

[e.g., survey by Mandi et al., JAIR'24]

Related work

Decision-making under uncertainty + machine learning + theory

Data-driven algorithm design

[e.g., Gupta, Roughgarden '16; Balcan, Nagarajan, [V](#), White, '17; Kleinberg et al., '17; Balcan, Dick, [V](#), '18; Garg, Kalai, '18; ...]

Algorithms with predictions

[e.g., book chapter by Mitzenmacher, Vassilvitskii, '20]

Calibration for decision-making

[e.g., Foster, Vohra, GEB'97; Zhao et al., NeurIPS'21; Kleinberg et al., COLT'23; Roth, Shi, EC'23; Noarov et al., ICML'25; ...]

Data-driven robust optimization

[e.g., Esfahani, Kuhn, '17; Bertsimas et al., '18; ...]

Decision-focused learning

[e.g., survey by Mandi et al., JAIR'24]

Ongoing/future work:
unification of these styles
(later in talk)

Additional related work

- Khodak et al. [NeurIPS'22]: Learning prediction reliability online
 - Similar in motivation, but not through lens of statistical UQ
- Sun et al. [ICML'24]: Algorithms with conformal ML predictions
 - We show calibration has key advantages over conformal methods
 - Especially helpful when predictions have high variance

Outline

1. Introduction
- 2. Background: Calibration**
3. Calibration for online algorithms
4. Conclusion and future directions

Calibration (binary target)

- Random variables (X, Y) with support $\mathcal{X} \times \{0,1\}$
- $f: \mathcal{X} \rightarrow [0,1]$ is **calibrated** if $\mathbb{P}[Y = 1 \mid f(x) = p] = p$
- E.g., rain prediction: Weather is rainy
 - 50% of days where $f(X) = 0.5$
 - 100% of days where $f(X) = 1$
- Let $T(X) = \mathbb{P}[Y = 1 \mid f(X)]$ (equals $f(X)$ if perfectly calibrated)



- Calibrated, **unsharp**:
 $f(X) = \mathbb{P}[Y = 1]$ for all X
- Calibrated, **sharp**:
 $f(X) = \mathbb{P}[Y = 1 \mid X]$ for all X

$$\underbrace{\mathbb{E} \left[(Y - f(X))^2 \right]}_{\ell_2 \text{ error}} = \underbrace{\text{Var}(Y)}_{\text{Uncertainty}} - \underbrace{\text{Var}(T(X))}_{\text{Sharpness}} + \underbrace{\mathbb{E} \left[(T(X) - f(X))^2 \right]}_{\text{Calibration error}}$$

ℓ_2 error

Uncertainty

Sharpness

Calibration error

Outline

1. Introduction
2. Background: Calibration
3. Calibration for online algorithms
 - i. **Rent-or-buy**
 - ii. Job scheduling
4. Conclusion and future directions

Online rent-or-buy

- Models buy-vs-rent under unknown horizon
- Balances upfront cost against ongoing rental expenses



Cloud:
on-demand vs
reserved instances



Equipment:
lease machinery vs
purchase outright



Mobility:
transit passes,
bike share

Ski rental problem

Prototypical online **rent-or-buy** decision-making problem

- Skier will ski for some **unknown** number of days $Z \in \mathbb{R}_+$
- Each day, decide to rent skis for \$1 or buy for one-time cost $\$b$
- **Goal:** Minimize total skiing cost
- Worst-case "breakeven" strategy:
Rent for b days, and if still want to ski, buy [Karlin et al. '01]

$$\text{Competitive ratio (CR)} := \frac{\text{ALG}}{\text{OPT}} = \frac{\text{amount algorithm pays}}{\min\{Z, b\}} \leq 2$$

Ski rental with predictions: Prior work

Algorithm with prediction of # skiing days Z [Kumar et al. NeurIPS'18]

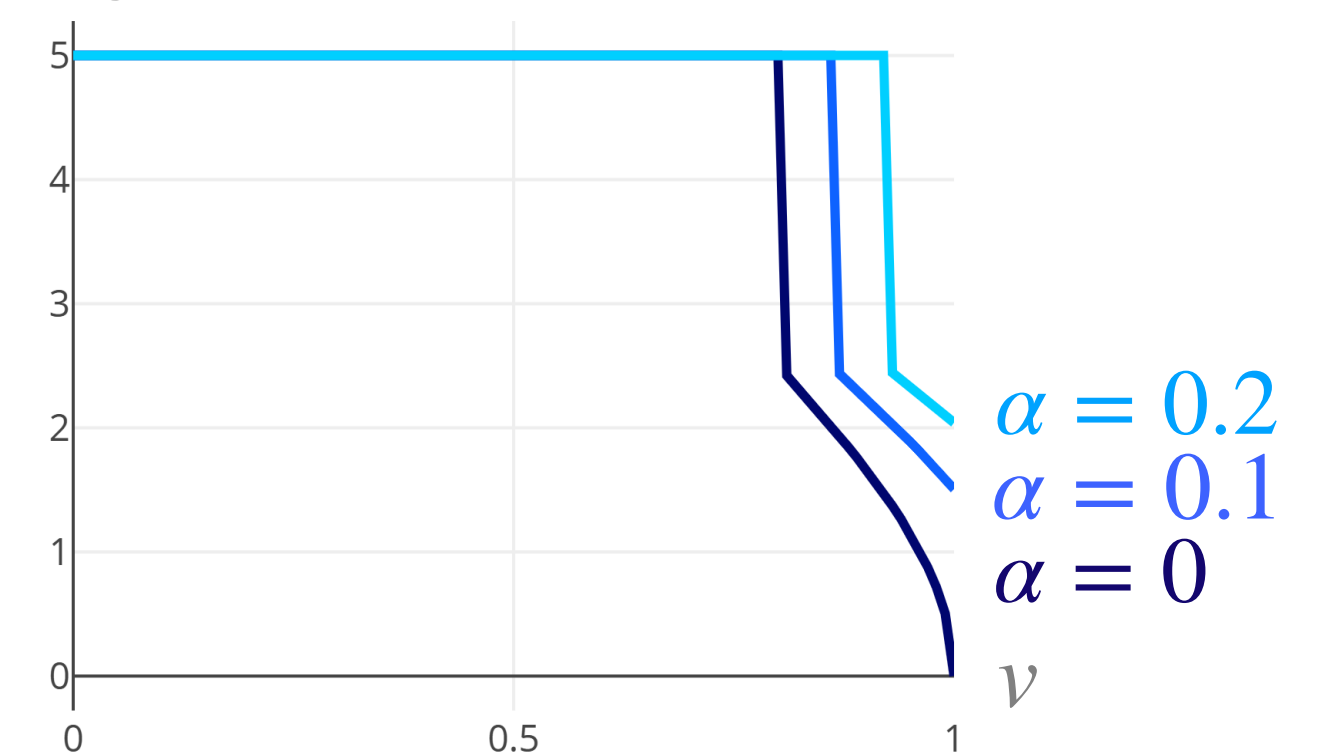
- Uses “trust” parameter $\lambda \in [0,1]$
 - $\lambda = 0$: fully trust predictor; $\lambda = 1$: don't trust predictor at all
- **Algorithm:** If prediction $\geq b$: buy on day $\lceil \lambda b \rceil$. Else, buy on day $\lceil b/\lambda \rceil$.
- **Consistency guarantee:** Perfect prediction yields $CR \leq 1 + \lambda$
 - If you knew to set $\lambda = 0$, recover optimal $CR = 1$
- **Robustness guarantee:** Any prediction yields $CR \leq 1 + \frac{1}{\lambda}$
 - If you knew to set $\lambda = 1$, recover best worst-case $CR = 2$
- **Our goal:** leverage calibration to encode trust/uncertainty

Algorithm with calibrated prediction

- $\mathcal{X}$ = skier features
- Predictor $f(X)$ of target $Y = 1(Z > b)$
Max calibration error $\alpha = \max_{v \in R(f)} |v - \mathbb{P}[Y = 1 \mid f(X) = v]|$
- **Algorithm:** given prediction $f(X) = v$ rent for $k(v)$ days

$$k(v) = \begin{cases} b & \text{if } v \leq \frac{4 + 3\alpha}{5} \\ b\sqrt{\frac{1 - v + \alpha}{v + \alpha}} & \text{else} \end{cases}$$

Days rented $k(v)$, ($b = 5$)

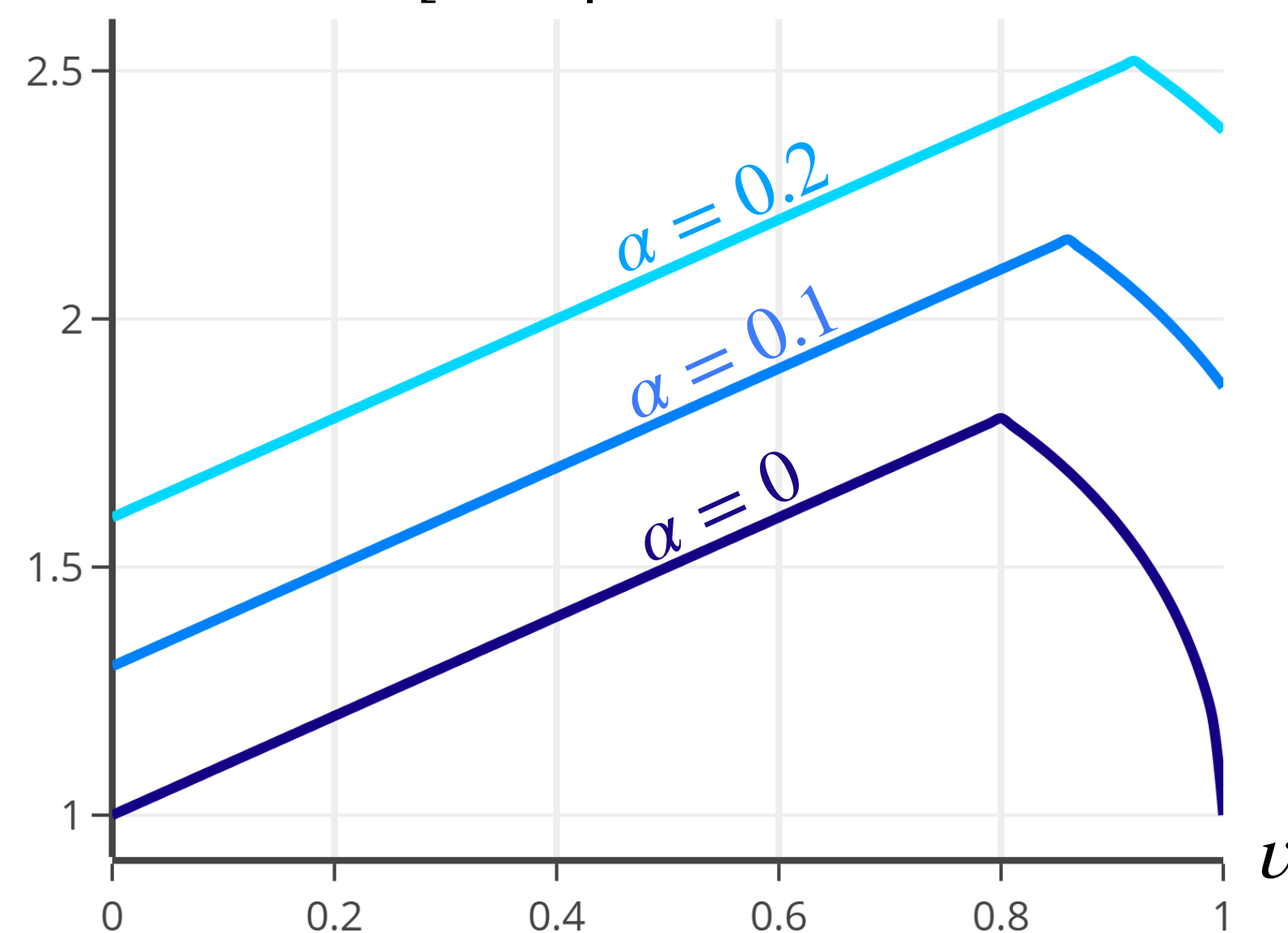


Ski rental: Main results

Prediction-wise bound:

$$\mathbb{E}[\text{CR} \mid f(X) = v] \leq 1 + 2\alpha + \min\left(v + \alpha, 2\sqrt{(v + \alpha)(1 - v + \alpha)}\right)$$

$\mathbb{E}[\text{CR} \mid f(X) = v]$



Ski rental: Main results

Prediction-wise bound:

$$\mathbb{E}[\text{CR} \mid f(X) = v] \leq 1 + 2\alpha + \min\left(v + \alpha, 2\sqrt{(v + \alpha)(1 - v + \alpha)}\right)$$

Lower bound: $\forall v$, exists distribution & calibrated predictor s.t.

$$\mathbb{E}[\text{CR} \mid f(X) = v] \geq 1 + \min\left(v, 2\sqrt{v(1 - v)}\right)$$

Global bound:

$$\mathbb{E}[\text{CR}] \leq 1 + 3\alpha + \min\left(\mathbb{P}[Z > b], 2\sqrt{\text{MSE}(f) + 3\alpha}\right)$$

✓ Lower MSE and calibration error lead to near-optimal CR

Outline

1. Introduction
2. Background: calibration
3. Calibration for online algorithms
 - i. Rent-or-buy
 - ii. Job scheduling**
4. Conclusion and future directions

Online job scheduling

Motivation [Cho et al. '22]: **radiologist's scheduling problem**

- One radiologist, backlog of imaging cases
- True case **urgency** initially **unknown**
 - Partial review reveals true urgency
- ML model provides **noisy** urgency probabilities
- Must choose processing order under uncertainty
- Goal: develop a simple, **robust scheduling policy**

Online job scheduling

- 1 machine to process n unit-length jobs
- Each job i has **unknown** high ($y_i = 1$) or low ($y_i = 0$) **priority**
 - Processing a θ -fraction of a job reveals its priority
- Jobs can be stored after partial processing
- Objective: Minimize weighted sum of completion times

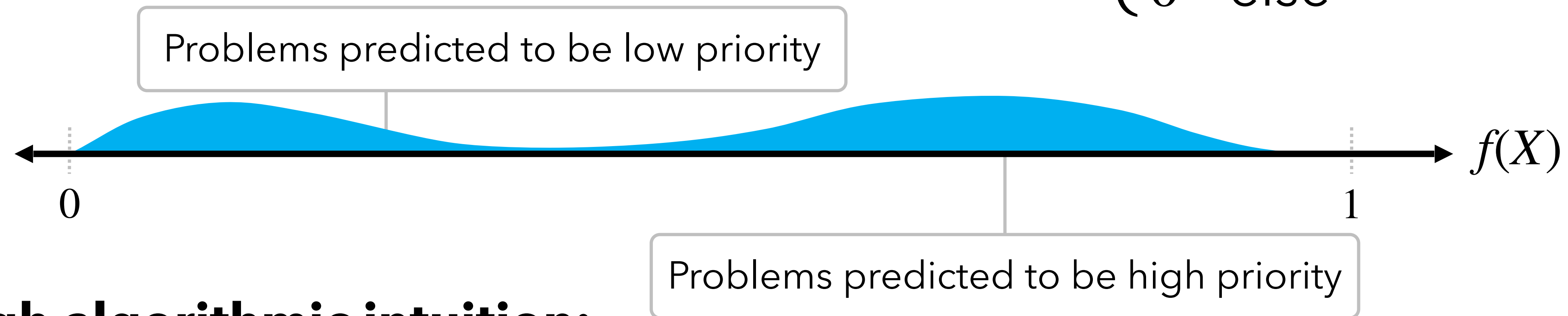
$$\sum C_i \cdot \omega_{y_i}$$

Completion time of job i

Cost per unit delay, with $\omega_1 > \omega_0 > 0$

Algorithmic intuition

Given job features X , predictor $f(X) \in [0,1]$ of $Y = \begin{cases} 1 & \text{if job is high priority} \\ 0 & \text{else} \end{cases}$



Rough algorithmic intuition:

Run jobs predicted to be high priority first, then low priority later, preemptively

If discover current is low-priority, stop, move to end of queue, and start new job

Importance of predictor sharpness

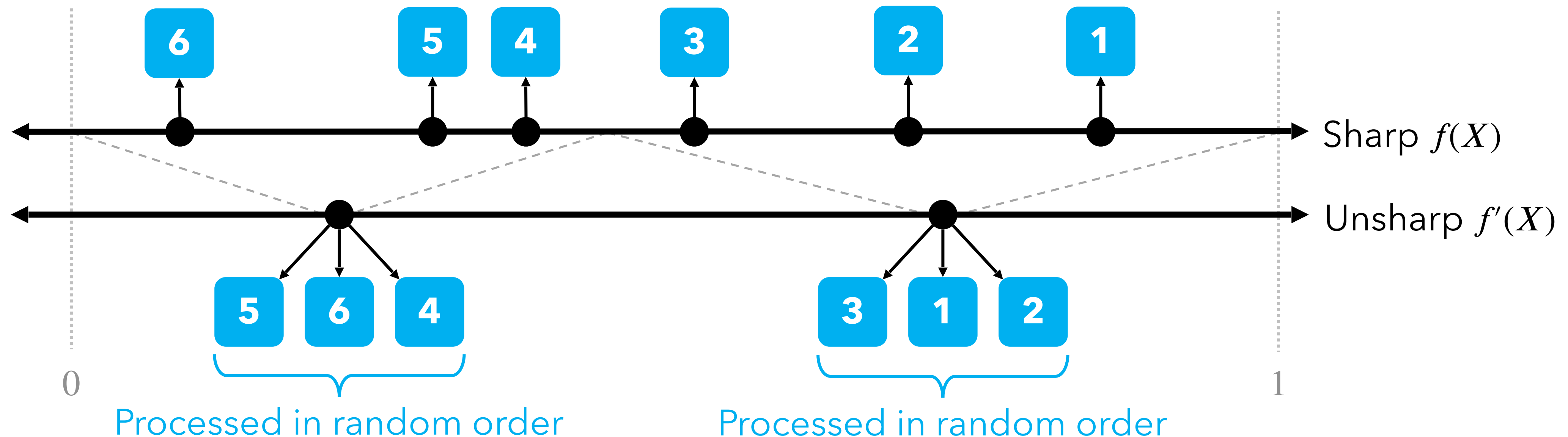
Key insight: **interchanges** are the primary source of **regret**

Weighted sum of completion times compared to optimal in hindsight

Importance of predictor sharpness

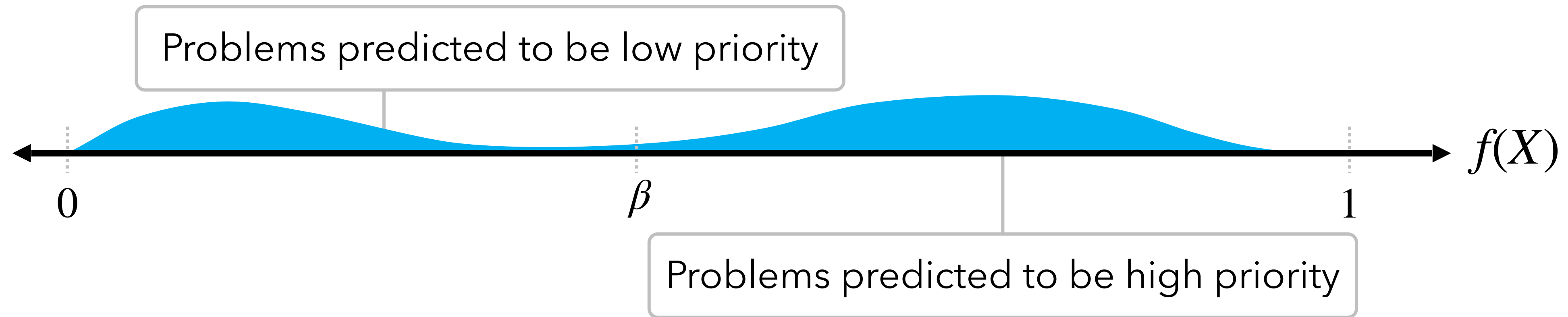
Key insight: **interchanges** are the primary source of **regret**

- Low priority job (partially) processed before high priority job
- **Sharp predictors** lead to **fewer interchanges**



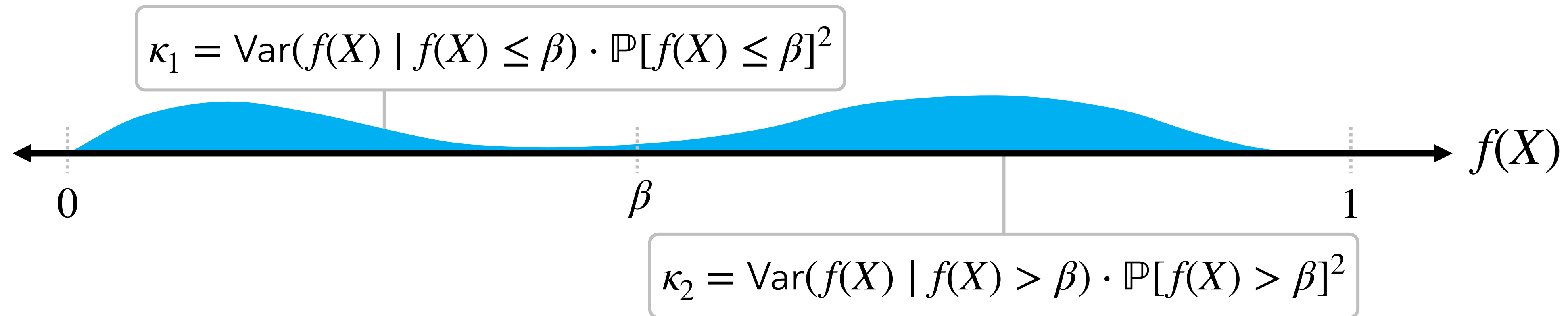
Importance of predictor sharpness

More formally



Importance of predictor sharpness

More formally



Thm: Let $\mathbb{P}[f(X) > \beta \mid Y = 0] = \epsilon_0$, $\mathbb{P}[f(X) \leq \beta \mid Y = 1] = \epsilon_1$

$$\mathbb{E}[\text{fraction of interchanged jobs}] \leq \underbrace{\text{Var}(Y)}_{\text{Inherent uncertainty}} (\underbrace{\epsilon_0 + \epsilon_1}_{\text{False positive/negative}}) - (\underbrace{\kappa_1 + \kappa_2}_{\text{Sharpness}})$$

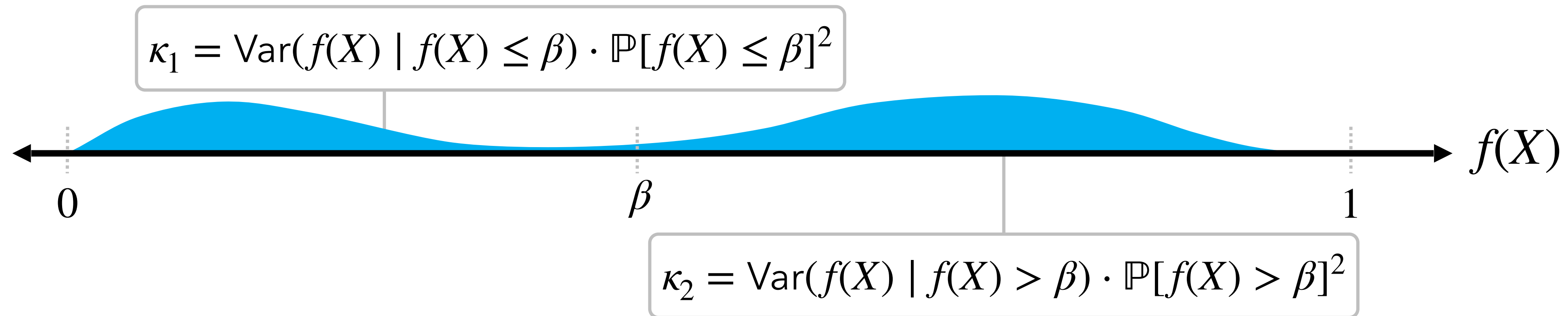
Inherent
uncertainty

False
positive/negative

Sharpness

Importance of predictor sharpness

More formally



Thm: Let $\mathbb{P}[f(X) > \beta \mid Y = 0] = \epsilon_0$, $\mathbb{P}[f(X) \leq \beta \mid Y = 1] = \epsilon_1$

$$\mathbb{E}[\text{fraction of interchanged jobs}] \leq \text{Var}(Y)(\epsilon_0 + \epsilon_1) - (\kappa_1 + \kappa_2)$$

(Eventual) corollary: bound on algorithm's regret (see paper)

Experiments: Sepsis triage

ML for predicting **sepsis onset** to improve early detection

Dataset of 110,204 hospital admissions

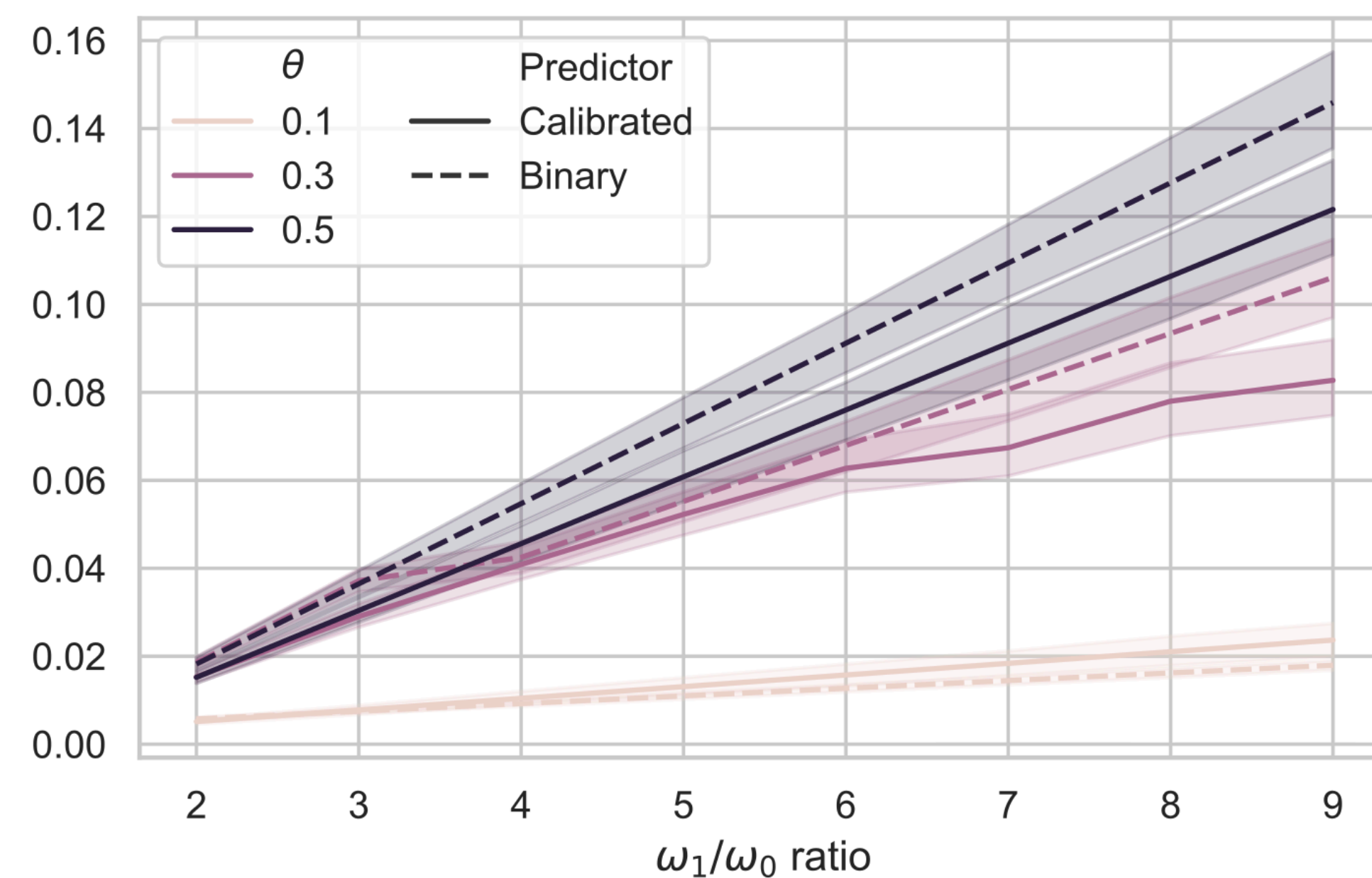
Base logistic regression
predictor (0.86 AUC)

*Unsharp
calibration* *Sharp
calibration*

"Binary"

"Calibrated"

Expected regret per job*



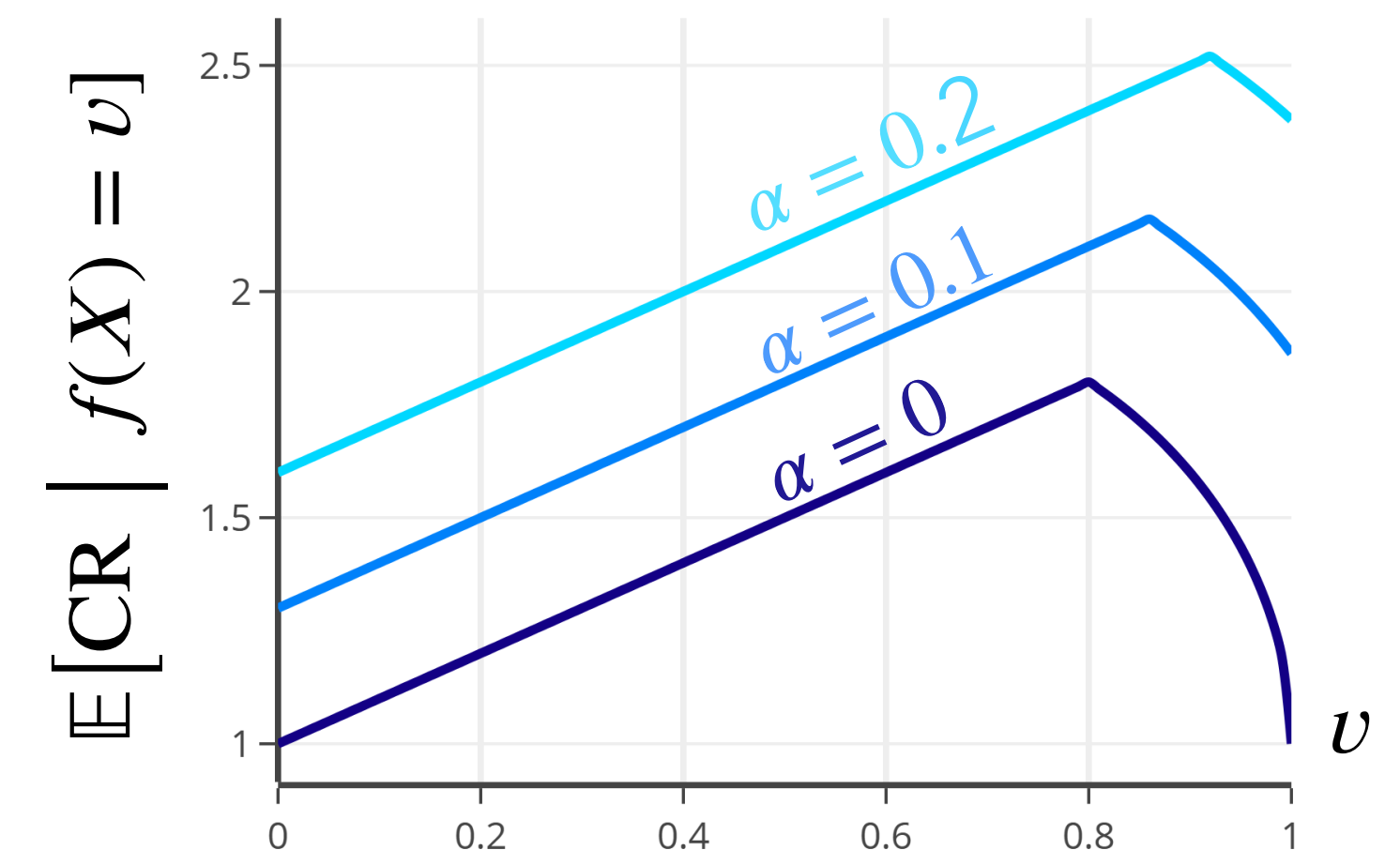
*Scheduling $n = 100$ patient reviews

Outline

1. Introduction
2. Background: Calibration
3. Calibration for online algorithms
- 4. Conclusion and future directions**

Conclusions

- **Calibration:** valuable tool for integrating ML into algorithms
- **Case studies:** ski rental and job scheduling
- Performance guarantees improve with calibration error
- Validated methods on real-world datasets



Ongoing directions

Connections to data-driven algorithm design

- Central question of **data-driven algorithm design**:
 - What components of algorithmic pipelines are learnable?
 - Algorithm parameters, uncertain aspects of the problem instance, ...
- **Connection to talk:** *max calibration error* is not efficient to optimize
- **Question:** what UQ methods are **learnable** and **algorithmically useful**?
- Connections to different ways to measuring mis-calibration
[E.g., Błasiok et al., '23; Kleinberg et al., '23; ...]

Algorithms with Calibrated Machine Learning Predictions

Ellen Vitercik, Stanford

ICML'25



Judy Hanwen Shen



Anders Wikum