Value alignment and control

- King Midas, paperclips and trolleys
- Measure what matters and manage tradeoff
- **Prediction versus optimization and control**
- Maintaining human oversight

Description, prediction, explanation, optimisation

Descriptive stats focus on summarizing properties of y,X y, X

Predictive models focus on labels on new examples X $y = f(X)$

Explanatory models focus on the structure of the learnt function $y = f(X)$

Prescriptive models wish to set X to achieve min/max y value $y = f(X)$

Description, prediction, explanation, optimisation

Histograms or clustering on the feature space y, X

Predict whether an image is that of a cat or a dog $y = f(X)$

Does this treatment improve the odds of survival of a patient? $y = f(X)$

What is the optimal distance to keep from the car in front of you? $y = f(X)$

Optimization via prediction

Consider a function f(x) that we wish to maximize, whose value we only know in certain locations:

$$\underset{x}{\operatorname{argmax}} f(x), \text{ given } y_i = f(x_i) \text{ for } i = 1, \ldots, n$$

For example, evaluating f(.) on a new example might require a real-world experiment, or running a expensive simulation (e.g., physics simulation). It is tempting to consider this as a prediction task:

$$\underset{x \in X}{\operatorname{argmax}} f_{\hat{\theta}}(x) \text{ where } \hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{n} L(f_\theta(x_i), y_i)$$

- Hard search problem (how do we generate new candidates X, and how many such candidates?)
- Hard extrapolation problem (how well does our predictor generalize to out-of-sample values?)
- Prediction task is optimized to solve for average accuracy, not for identifying the maximum/minimum

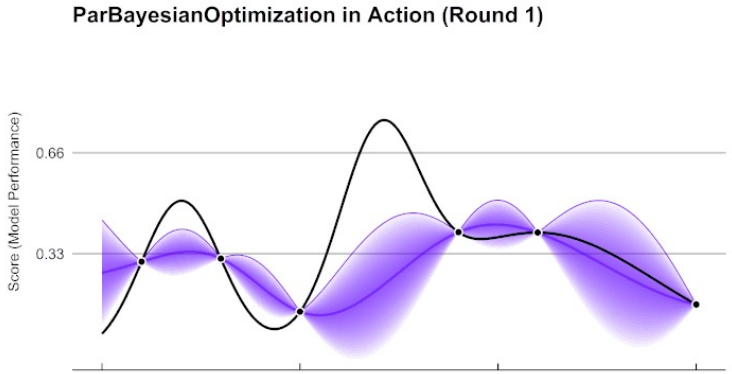
Optimization via prediction

$$\underset{x \in X}{\operatorname{argmax}} f_{\hat{\theta}}(x) \text{ where } \hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{n} L(f_\theta(x_i), y_i)$$

- **Search**: exploit smoothness or gradient information in the estimated function to direct the search towards areas likely to achieve a maximum
- **Extrapolation:** try some form of active learning, where in areas of maximum uncertainty we are requesting extra function evaluations (epistemic humility)
- **Training the right predictor:** smooth functional approximators like Gaussian Processes might be more appropriate than, say, a random forest.

Exploration-exploitation

A core concept in Bayesian Optimization is that of trading off exploration of parts of the space that you know little about, and exploitation of your current best guess about where the maximum is - like always going to the same restaurant after only having tried a few.



Exploration-exploitation

The exploration-exploitation tradeoff is fundamental to all **sequential decision-making processes**. Examples that we will consider in later parts of the course include:

- Upper Confidence Bound (UCB)
- Thompson Sampling

Variations of these techniques can be used both in simpler Multi-Armed Bandit contexts and in the more general Reinforcement Learning context (to be defined soon).

Why should you care?

Production models are sequential decision-makers

A model ”in production” is in fact just one step in a sequential decision-making process:

- They might create a selection mechanism (e.g., a pulse oximeter failing on dark skin will stop being used by Black patients, removing them from future training sets)
- They might result in changing behavior on the part of the users (e.g., the use of an ML system for cybersecurity might result in evasive behavior from the attackers).
- The observed performance of an ML model might trigger its discontinuation, or a change in the loss function thereafter (e.g., a switch to a more risk-averse position).

Ignoring the broader decision-making context of a predictive model can propagate bias, is only locally value aligned and increases risk of long-term adverse consequences.

Summary

- Most basic machine learning pipelines are focused on prediction, and designed to optimize some version of accuracy (minimize average error per example).
- Nevertheless, many real-world tasks are in fact optimization tasks. Including this fact into our design thinking makes for more efficient algorithms.
- Sequential decisions introduce further tradeoffs between exploration and exploitation. Multiple frameworks exist to model that, ranging from bandits to reinforcement learning.
- Often the broad context in which a model operates is in fact one of sequential decision making (with or without a human in the loop). Ignoring that structure introduces risk of longer-term unanticipated consequences, such as self-reinforcing confirmation bias.