The Cloud Hunter's Problem

An Automated Decision Algorithm to Improve the Productivity of Scientific Data Collection in Stochastic Environments

Arthur Small

based on work with JASON B. STEFIK, JOHANNES VERLINDE, NATHANIEL C. JOHNSON

7 Feb 2020

Hello. . .

IS THERE ANYBODY *in* THERE?

Theme: Data-driven decision making

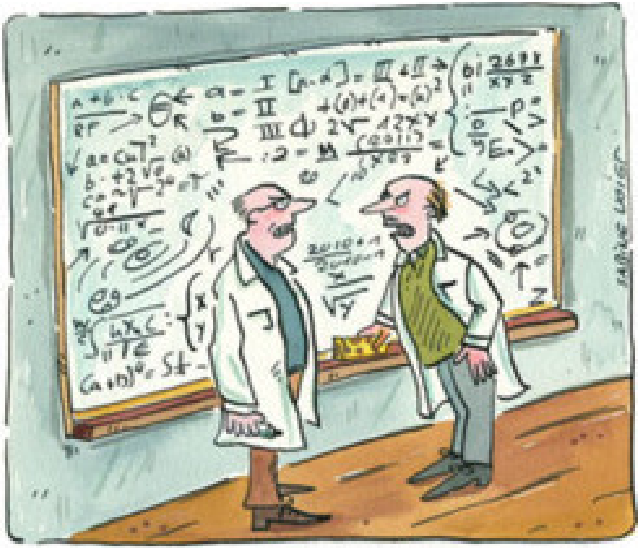
Introduction: The problem

Introduction: The problem

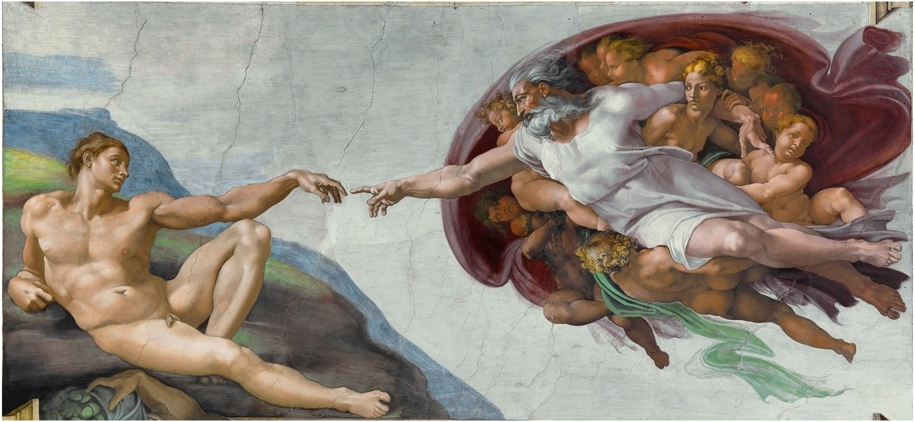
The Cloud Hunter's basic problem is to decide how to allocate a limited budget of flight time over the course of a field season.

Decisions must be made 1 day ahead, based on imperfect day-ahead forecasts of whether conditions are good or bad.

How they make their decisions now:



What we offer instead:



Formal model of the decision problem

Model setup

D : length of the field season in days

$F \leq D$: number of flights in the Cloud Hunter's budget.

$d = D, \ldots, 1$: index of dates

$f = F, \ldots, 1$: index of flights remaining in budget

X_d : binary random state variable indicating quality of conditions for data collection:

- $X_d = 1$ if conditions on date d are good, 0 otherwise
- x_d : realized value of X_d

a_d : binary control variable

- $a_d = 1$ iff they opt to fly on date d, 0 otherwise

$\mathbf{X} =< \mathbf{X_D}, ..., \mathbf{X_1} >$: conditions on dates $d = D, \ldots, 1$

$\mathbf{a} =< \mathbf{a_D}, ..., \mathbf{a_1} >$: actions chosen on dates $d = D, \ldots, 1$

Objectives

Choose **a** to

$$max_{\mathbf{a}} E[\mathbf{a} \cdot \mathbf{X}]$$

Decision taken on basis of day-ahead forecast

Before taking each decision, decision-maker receives a forecast signal $s_d \in \mathbb{S}$.

Based on a calibrated prediction model, can map this signal to a probability of good conditions:

$$p(s) = \Pr\{X_d = 1 | s_d = s\}$$

.

Distribution of forecast signals

Distribution of forecast signals

$\pi(s)$: unconditional probability that forecast will take value s.

$\pi(\cdot)$ defines a probability distribution over the set $\mathbb{S}$ of possible forecast signals.

Intertemporal optimization via dynamic programming

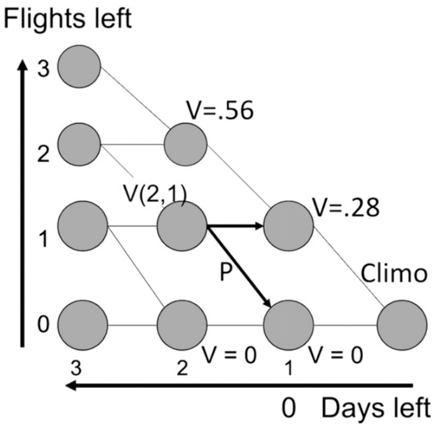


Figure 1: Graphical representation of the decision algorithm.

Probabilistic forecasting of favorable conditions using self-organizing maps

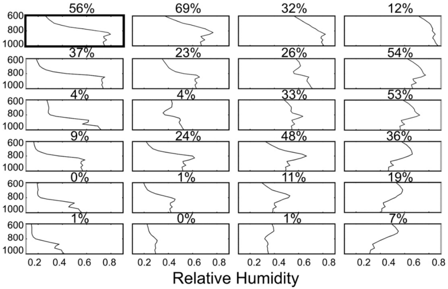


Figure 2: 6 X 4 SOM grid for relative humidity profiles

13.3%	20.6%	4.4%	3.7%
1.5%	8.1%	1.5%	4.4%
0%	3.7%	4.4%	11.8%
2.9%	3.7%	7.4%	6.6%
0%	0%	0.7%	1.5%
0%	0%	0%	0%

Figure 3: Conditional probability distribution of SOM state realizations following a forecast of SOM state 1

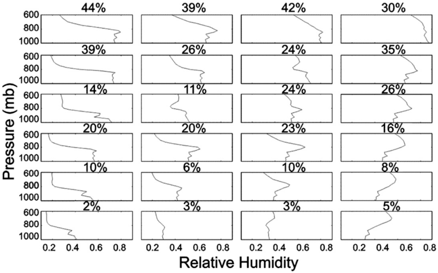


Figure 4: Estimated probabilities of good conditions for data collection, as a function of day-ahead SOM forecast

Results

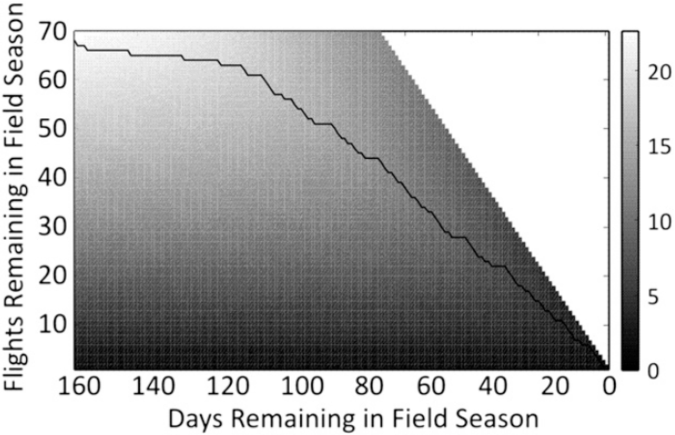


Figure 5: Computed values for the value function $V(d, f)$

Dark line : simulated sequence of flight decisions

- Diagonal movements = fly dates, horizontal movements = no-fly dates

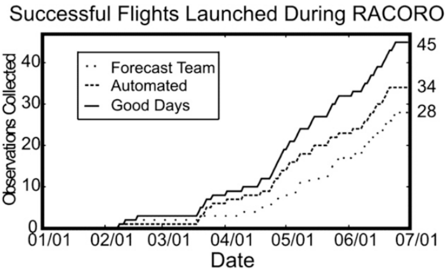


Figure 6: Results: the algorithm's simulated performance during 2009 field season, compared with realized performance of heuristic decision procedure.

The algorithm achieves a 21% increase in the number of successful flights.

	Heuristic procedure	Automated algorithm
Flights launched	56	66
Successes	28	34
Type I errors	28	32
Type II errors	17	11

Figure 7: Summary of outcomes

Successes are flights launched on days with good conditions.

Type I errors are decisions to fly only to find no clouds.

Type II errors are decisions to stand down only to find that the desired conditions existed.