# Learning from controlled sources

Onno Zoeter

Industry Director Mercury Machine Learning Lab
Booking.com

Mercury
machine learning lab

Universiteit van Amsterdam

Booking.com

TU Delft

# The Mercury Machine Learning Lab

Mercury
machine learning lab

Theme 1: **Learning from Controlled Sources**

Information Retrieval    Causality    Reinforcement Learning

| | |
|---|---|
| *University of Amsterdam*: | Prof. Joris Mooij, Prof. Maarten de Rijke |
| | Philip Boeken, Philipp Hagar |
| *TU Delft*: | Prof. Frans Oliehoek, Prof. Matthijs Spaan |
| | Oussama Azizi, Davide Mambelli |

Theme 2: **Natural Language processing – Multi-task learning**

| | |
|---|---|
| *University of Amsterdam*: | Prof. Wilker Aziz, Prof. Ivan Titov |
| | Pedro Ferreira |

# Learning from controlled sources

Building and using a model $P(y|x)$ is "easy"

$$
\begin{aligned}
P(\theta|\{(x_i, y_i)\}) &= \frac{\prod_i P(y_i|x_i, \theta)P(\theta)}{\prod_i P(y_i|x_i)} \\
a^* &\in \arg\max_a \sum_{y_{n+1}} U(a, y_{n+1})P(y_{n+1}|x_{n+1}) \\
P(y_{n+1}|x_{n+1}) &= \int P(y_{n+1}|x_{n+1}, \theta)P(\theta|\{(x_i, y_i)\}d\theta
\end{aligned}
$$

**However in many real-world applications the problem is richer, *because* the model is used:**

- The training data is biased because the current model filters out cases
- The training data represents environment + old model acting
- Because the model acts, the environment changes
- $\cdots$

# Example: biased training data

Online payment application:
$y \in \{fraud, valid\ payment\}$
$x$: observed features at decision time

If $P(y = fraud|x)$ is low: accept, $s(x) = 1$

Filtering: if $s(x) = 0$ then $y$ unobserved

| | $X$ | $S$ | $Y$ |
|---|---|---|---|
| 1 | $x_1$ | 1 | 0 |
| 2 | $x_2$ | 0 | ? |
| 3 | $x_3$ | 1 | 1 |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $n$ | $x_n$ | 0 | ? |

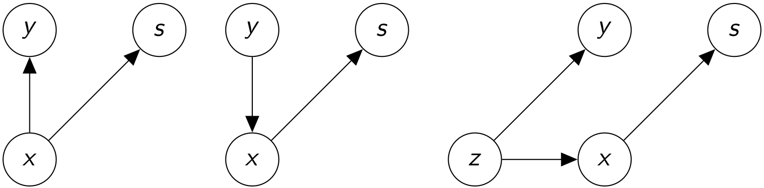
**Question**

Can you learn $P(y|x)$ from this data?

If so, do you need correction factors?

# s-recoverability

**Idea**: model data generating process and selection process $s(x)$ explicitly

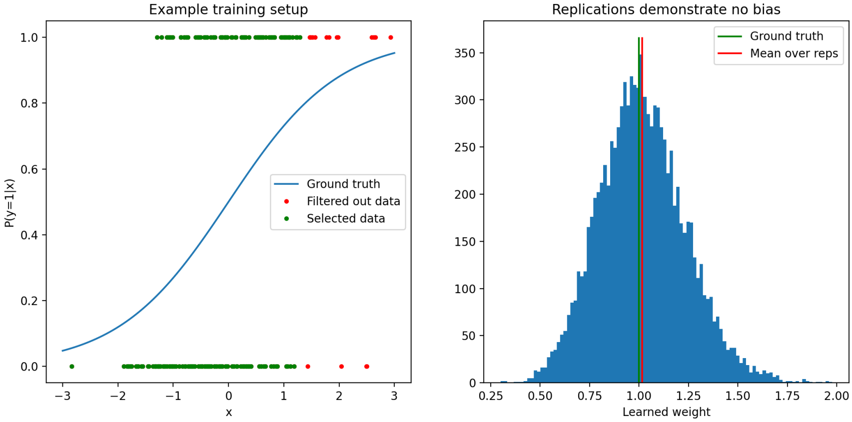



Using d-separation, in all models $y \perp s|x$ so $P(y|x, s(x) = 1) = P(y|x)$

$\{(x_i, y_i)|s(x_i) = 1\}$ can be used to learn $P(y|x)$

Bareinboim, Elias, and Judea Pearl. "Controlling selection bias in causal inference." PMLR, 2012.

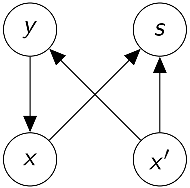
# Filtering training data on inputs $x$



# Adding complexity

More difficult:

Heuristic $s'(x')$ filters as well




**Question**

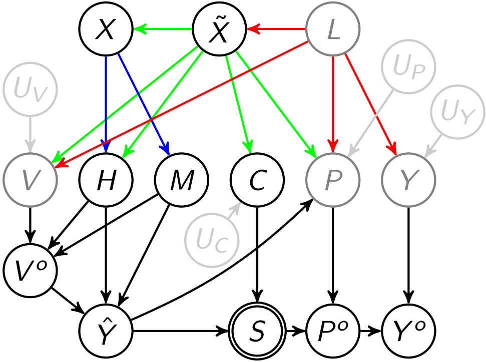
Can we learn $P(y|x)$ from this biased data?

**Idea**: conditioning on $x$ and $x'$ makes $y$ and $s$ independent.

$$\begin{aligned} P(y|x) &= \sum_{x'} P(y, x'|x) \\ &= \sum_{x'} P(y|x, x')P(x'|x) \\ &= \sum_{x'} {\color{green}P(y|x, x', s(x) = 1)}{\color{brown}P(x'|x)} \end{aligned}$$

We can learn ${\color{green}P(y|x, x', s(x) = 1)}$ from the filtered data and ${\color{brown}P(x'|x)}$ if we observe $(x, x')$ always

# A realistic situation



# Discussion

## Learning from controlled sources

- Actual use of ML models leads to a richer problem than supervised learning
- Partial solutions already exist in IR, causality, RL, and related fields.

  Today's example: dealing with controllers that filter a data source
- The Mercury Machine Learning Lab will collect and connect existing results
- create what is missing, and
- work towards a general problem statement and a toolbox of practical algorithms