

Autocalibration & Insurance Pricing

Arthur Charpentier (UQAM)

with Michel Denuit (UCL) & Julien Trufin (UCB)

ASTIN Colloquium 2021

Agenda

- ▶ On insurance pricing
- ▶ What is a model ?
- ▶ Bias of a model
- ▶ Correcting from bias
- ▶ Application
- ▶ Theoretical properties



Demetri @PhDemetri · 13h

All this talk about XGboost prompted me to try it again on some toy datasets I have laying around.

The long and the short of it is: XGBoost results in a better AUC than my logistic regression (99.7 v 87) but XGB is so poorly calibrated it doesn't make sense to trust the probs

7

2

42



Demetri @PhDemetri · 13h

Calibration is a thing I never see people talk about in data science. We should really care about calibration more than we do.

2

1

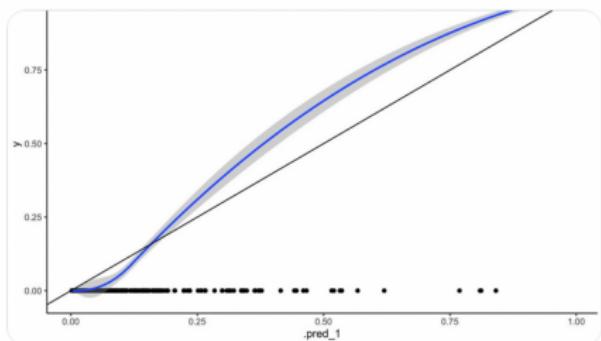
24



Demetri @PhDemetri · 13h

Calibration for the xgboost model.

Yuck



3

2

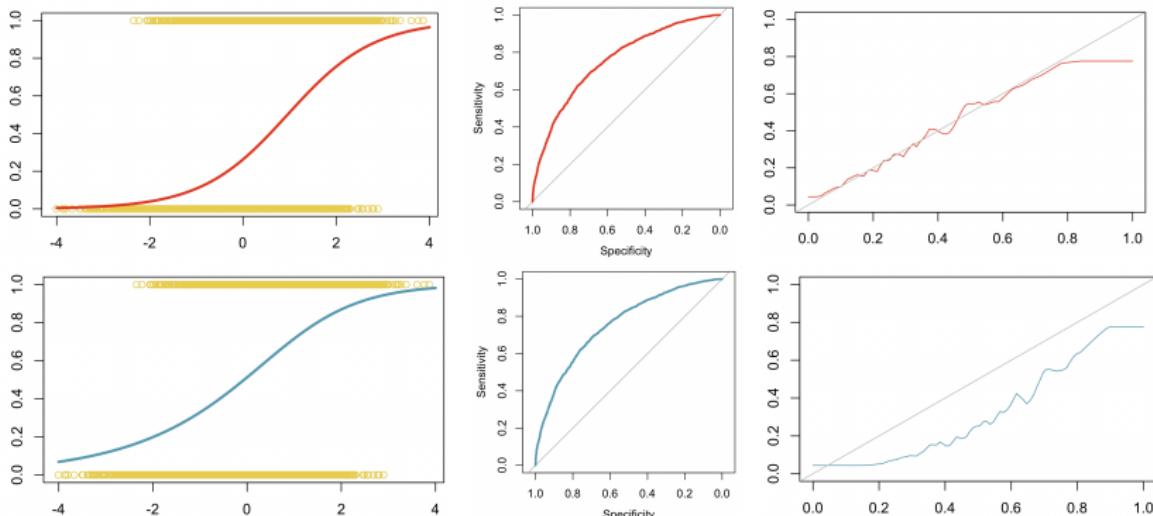
11



Machine Learning & Accuracy Measures (AUC = 0.7565)

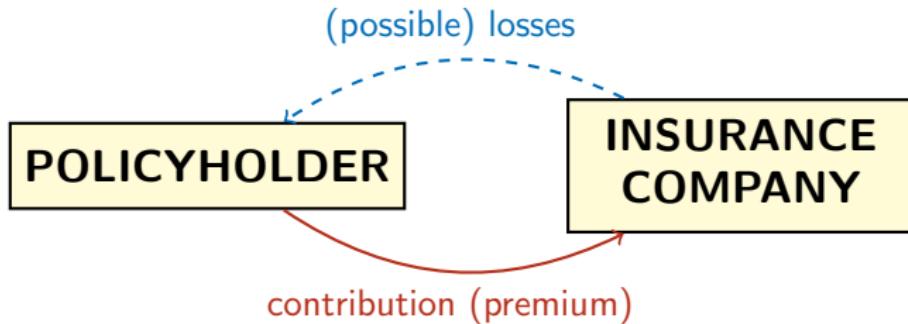
"XGBoost results in a better AUC than my logistic regression but XGBoost is so poorly calibrated it doesn't make sense to trust the probs "

- ▶ (x_i, y_i) (where $y_i \in \{0, 1\}$) with $\hat{\pi}_1(x)$ or $\hat{\pi}_2(x)$ ($= \sqrt{\hat{\pi}_1(x)}$)
- ▶ ROC curve of $\hat{\pi}_1$ or $\hat{\pi}_2$ are identical (and same AUC)
- ▶ $\mathbb{E}[Y|\hat{\pi}_1(X) = s]$ or $\mathbb{E}[Y|\hat{\pi}_2(X) = s]$,
ie. regression of y_i on $\pi(x_i)$ (expected? $\mathbb{E}[Y|\hat{\pi}(X) = s] \sim s$)



Insurance

“Insurance is the contribution of the many to the misfortune of the few”



The “*contribution*” is obtained using predictive [models](#)
(interpretability / black box / etc issues)

A model, $m : \mathbf{x} \mapsto y$

To train / estimate a model m ,
we need a dataset, i.e. a collection
of observations (\mathbf{x}_i, y_i)

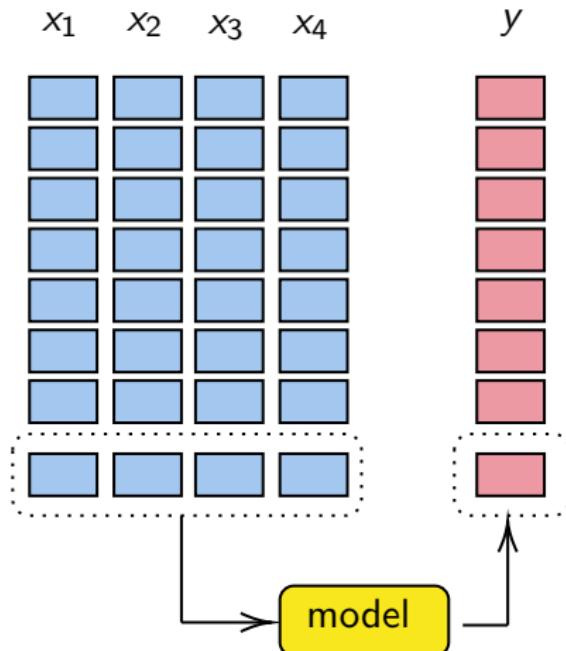
Usually y_i denotes the annual loss

$$y_i = \sum_{j=1}^{n_i} z_{i,j}$$

n is the annual frequency

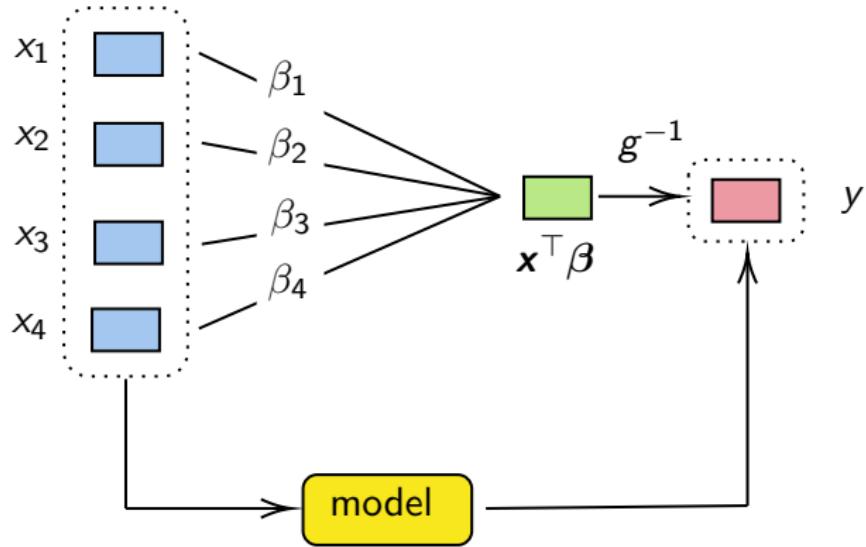
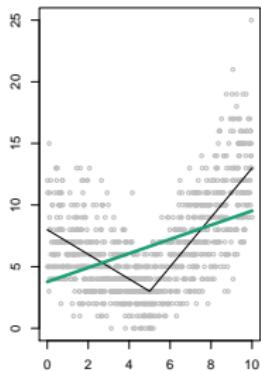
z is the cost of a single claim
(see Tweedie models)

To illustrate, y is a counting variable



A model, $m : \mathbf{x} \mapsto y$ (GLM)

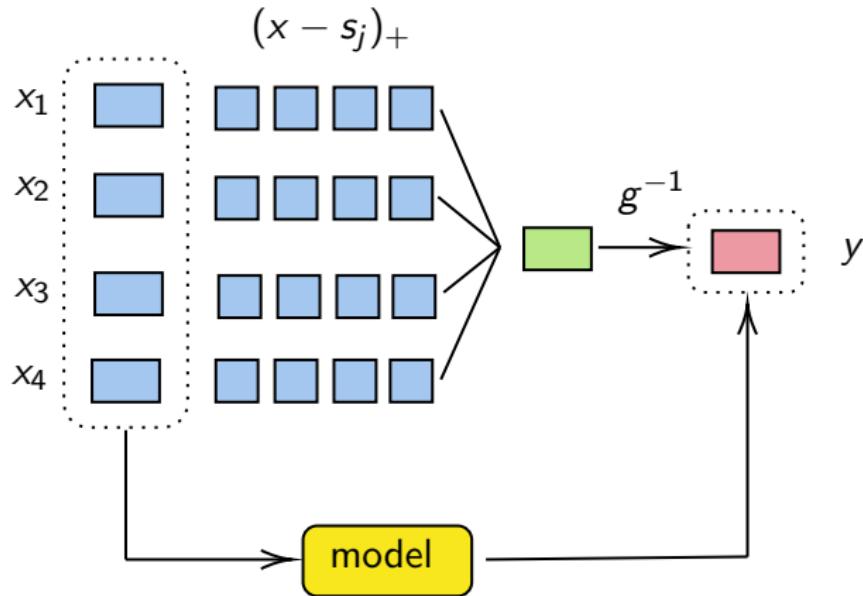
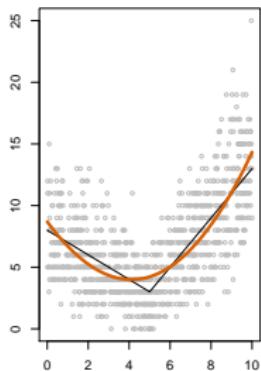
Poisson regression



$$m(\mathbf{x}) = g^{-1}(\mathbf{x}^\top \boldsymbol{\beta}) = g^{-1} \left(\sum_{j=1}^p \beta_j x_j \right)$$

A model, $m : \mathbf{x} \mapsto y$ (GAM)

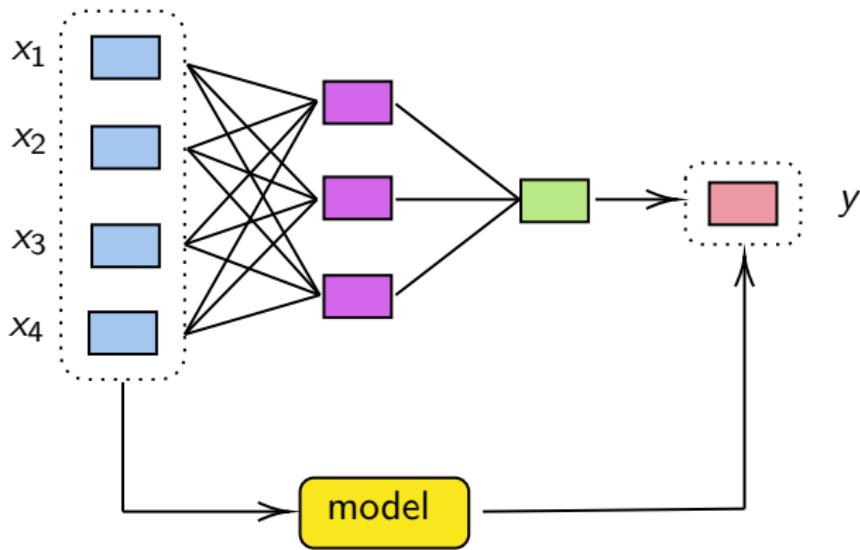
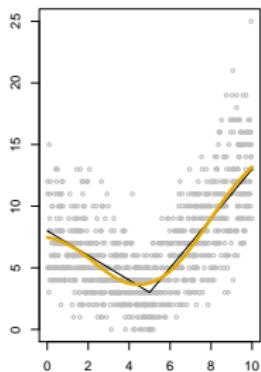
Poisson regression



$$m(\mathbf{x}) = g^{-1} \left(\sum_{j=1}^p \beta_j \psi_j(x_j) \right), \text{ where } \psi_j(x) = \sum_{k=1}^5 \alpha_{j,k} (x - s_{j,k})_+$$

A model, $m : \mathbf{x} \mapsto y$ (neural nets)

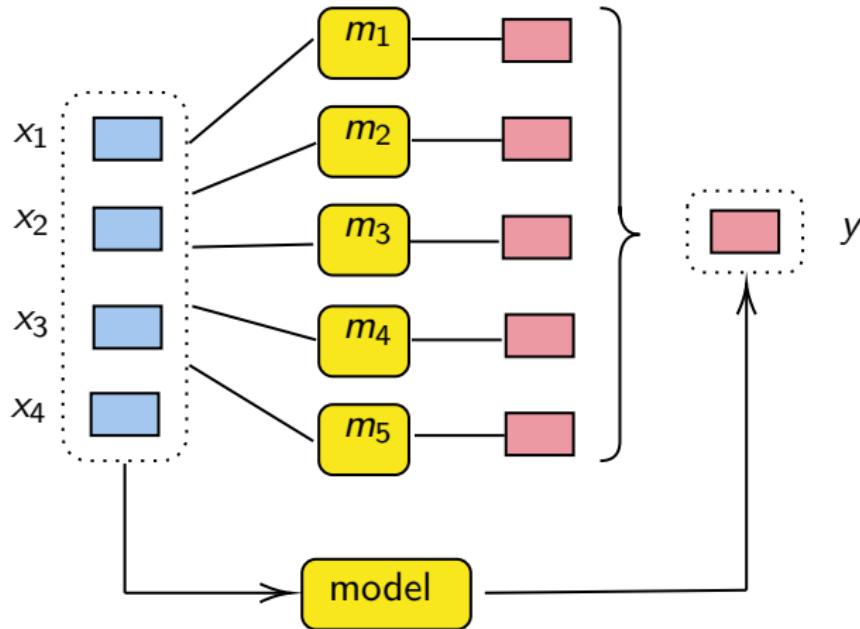
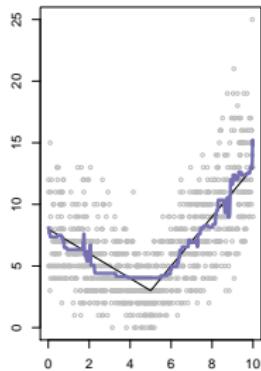
Poisson loss



$$\text{E.g. } m(\mathbf{x}) = \sum_{j=1}^3 \omega_{1:j} h(\mathbf{x}^\top \omega_{2:j})$$

A model, $m : \mathbf{x} \mapsto y$ (ensemble parallel, bagging)

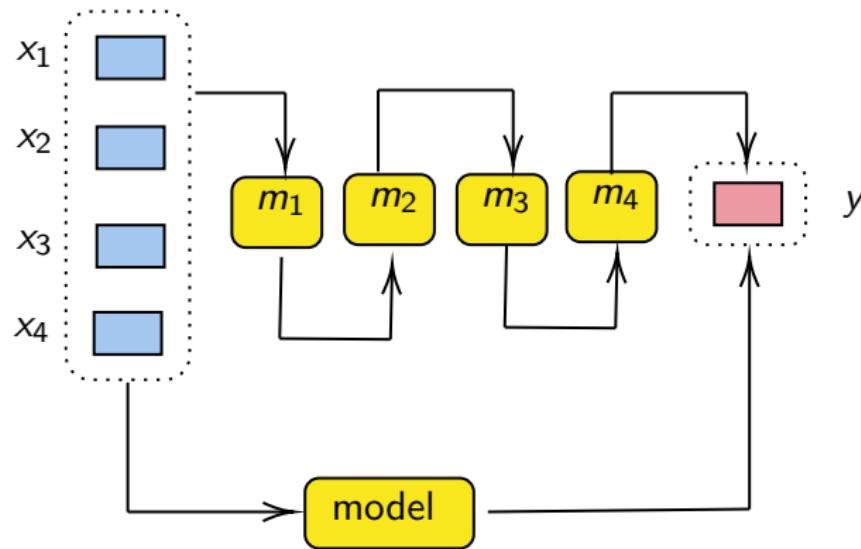
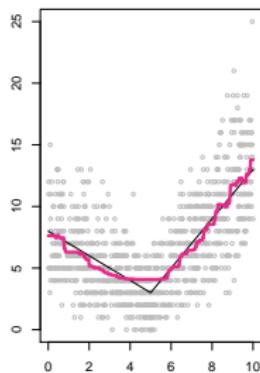
Random forest



$$\text{E.g. } m(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B m_b(\mathbf{x}) \text{ where } m_b(\mathbf{x}) = \sum_{k=1}^K \alpha_{b,k} \mathbf{1}(\mathbf{x} \in \mathcal{X}_{b,k})$$

A model, $m : \mathbf{x} \mapsto y$ (ensemble sequential, boosting)

Boosting



$$m(\mathbf{x}) = \sum_{t=1}^T m_t(\mathbf{x}) \text{ where } m_t \text{ is a (weak) model on } y_i - m_{t-1}(\mathbf{x}_i)$$

(residuals from the previous step) such as (not too deep) trees

Not a model, $\hat{m} : \mathbf{x} \mapsto y$ (local regression)

To approximate $\mathbb{E}[Y]$ use

$$\hat{m} = \underset{\mu \in \mathbb{R}}{\operatorname{argmin}} \left\{ \sum_{i=1}^n \frac{1}{n} [y_i - \mu]^2 \right\}$$

To approximate $\mathbb{E}[Y | \mathbf{X} = \mathbf{x}]$, use

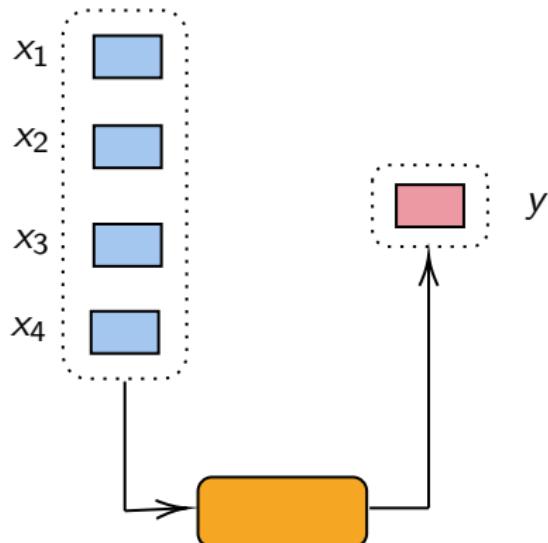
$$\hat{m}(\mathbf{x}) = \underset{\mu \in \mathbb{R}}{\operatorname{argmin}} \left\{ \sum_{i=1}^n \omega_i(\mathbf{x}) [y_i - \mu]^2 \right\}$$

where, see [Loader \(1999\)](#),

$$\omega_i(\mathbf{x}) \propto k_\alpha (\|\mathbf{x} - \mathbf{x}_i\|)$$

or k nearest neighbors indicator...

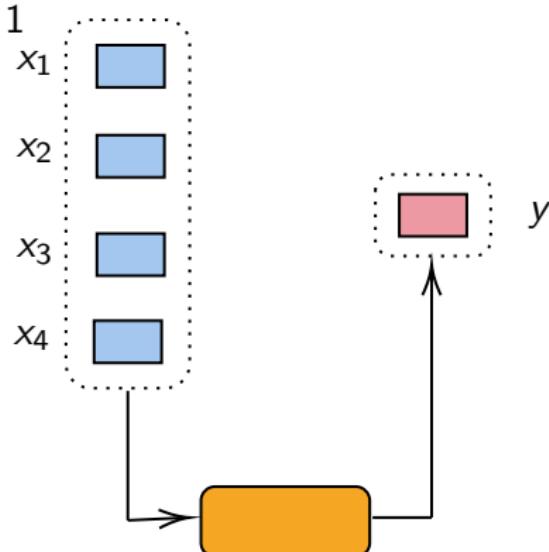
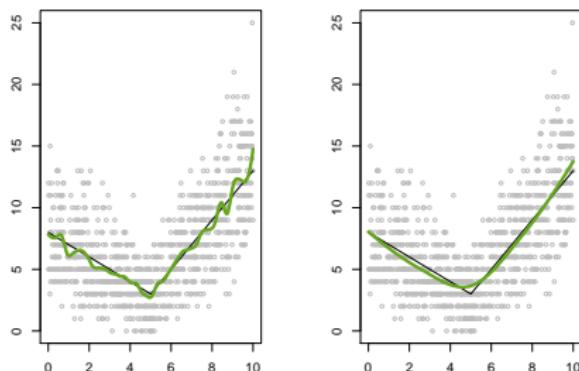
Behaves poorly in high dimension (but efficient in dimension 1)



Not a model, $\hat{m} : x \mapsto y$ (local regression)

$$\hat{m}(x) = \sum_{i=1}^n \omega_i(x) y_i, \text{ with } \sum_{i=1}^n \omega_i(x) = 1$$

see `locfit` in R



(depends on a bandwidth parameter α)
Provides a local estimate of $\mathbb{E}[Y|X = x]$ on a neighborhood of x

GLM, Bias, & Economic Interpretation

For GLMs, $f(y_i) = \exp\left(\frac{y_i\theta_i - b(\theta_i)}{\varphi} + c(y_i, \varphi)\right)$

The natural parameter for y_i : θ_i

Prediction for y_i : $\hat{y}_i = \mu_i = \mathbb{E}(Y_i) = b'(\theta_i)$

Score associated with y_i : $\eta_i = \mathbf{x}_i^\top \beta$

Link function : g such that $\eta_i = g(\mu_i) = g(b'(\theta_i))$

$$\log \mathcal{L}(\boldsymbol{\theta}, \varphi, \mathbf{y}) = \sum_{i=1}^n \log \mathcal{L}_i = \sum_{i=1}^n \left[\frac{y_i\theta_i - b(\theta_i)}{\varphi} + c(y_i, \varphi) \right]$$

First order conditions: $\frac{\partial \log \mathcal{L}_i}{\partial \beta_j} = \frac{\partial \log \mathcal{L}_i}{\partial \theta_i} \cdot \frac{\partial \theta_i}{\partial \mu_i} \cdot \frac{\partial \mu_i}{\partial \eta_i} \cdot \frac{\partial \eta_i}{\partial \beta_j}$

$$\frac{\partial \log \mathcal{L}_i}{\partial \theta_i} = \frac{y_i - \mu_i}{\varphi}, \quad \frac{\partial \theta_i}{\partial \mu_i} = \left(\frac{\partial \mu_i}{\partial \theta_i} \right)^{-1} = \frac{1}{b''(\theta_i)} = \frac{1}{V(\mu_i)}$$

$$\frac{\partial \eta_i}{\partial \beta_j} = \frac{\partial \mathbf{x}_i^\top \beta}{\partial \beta_j} = x_{i,j}, \quad \frac{\partial \mu_i}{\partial \eta_i} = \left(\frac{\partial \eta_i}{\partial \mu_i} \right)^{-1} = (g'(\mu_i))^{-1}$$

GLM, Bias, & Economic Interpretation

With canonical link $g_\star = b'^{-1}$, i.e. $\eta_i = \theta_i$,

$$\mathbf{X}^\top(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0}, \text{ where } \hat{\mathbf{y}} = \boldsymbol{\mu}$$

so, if there is an intercept, $\mathbf{1}^\top(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0}$, i.e. $\sum_{i=1}^n y_i = \sum_{i=1}^n \hat{y}_i$

which is the empirical version (training dataset) of $\mathbb{E}[Y] = \mathbb{E}[\hat{Y}]$
(see logistic regression or Poisson with log-link)

But more generally, the first order condition is

$$\mathbf{X}^\top \mathbf{W} \Delta (\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0},$$

where $\mathbf{W} = \text{diag}((V(\mu_i)g'(\mu_i)^2)^{-1})$ and $\Delta = \text{diag}(g'(\mu_i))$.

But usually not an important issue in ML classification problems

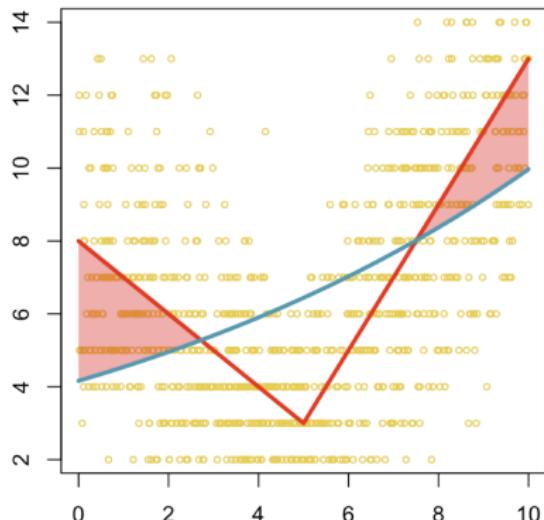
GLM, Bias, & Economic Interpretation

Model $\hat{\pi}$ is **globally unbiased** if $\mathbb{E}[\hat{\pi}(\mathbf{X})] = \mathbb{E}[Y]$, $\sum_{i=1}^n \hat{y}_i = \sum_{i=1}^n y_i$

Model $\hat{\pi}$ is **locally unbiased** if $\mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = s] = s$

GLM $\hat{\pi}$ globally unbiased,
but possibly **locally biased**
Major economic impact

- ▶ $\hat{\pi}(\mathbf{x}) < \mu(\mathbf{x})$
attractive, but underpriced
- ▶ $\hat{\pi}(\mathbf{x}) > \mu(\mathbf{x})$
not attractive

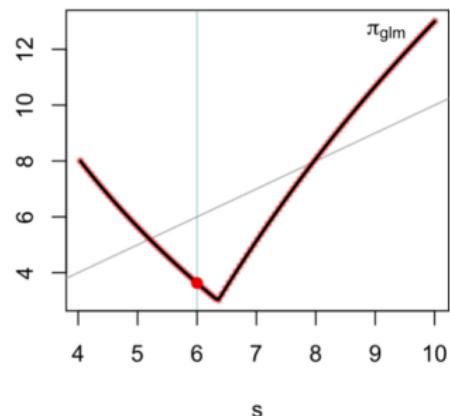
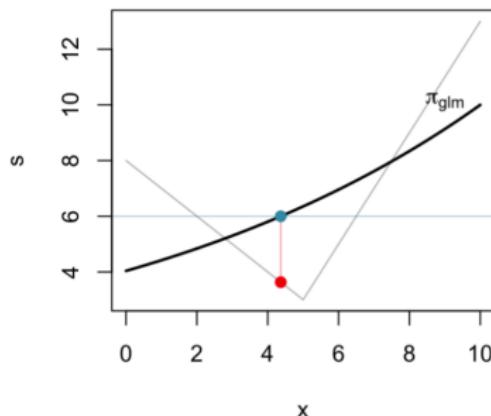


Natural idea: plot $s \mapsto \mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = s]$ “ $= \mu(\hat{\pi}^{-1}(s))$ ”

Computing $\mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = s]$

True model $\mu(x)$ and GLM model $\hat{\pi}(x)$

- ▶ select s , e.g. $s = 6$
- ▶ compute $x = \hat{\pi}^{-1}(s)$ ($\hat{\pi}$ is strictly increasing), here $x = 4.2$
- ▶ compute $\mu(\hat{\pi}^{-1}(s))$, here 3.8
- ▶ plot $(s, \mu(\hat{\pi}^{-1}(s)))$

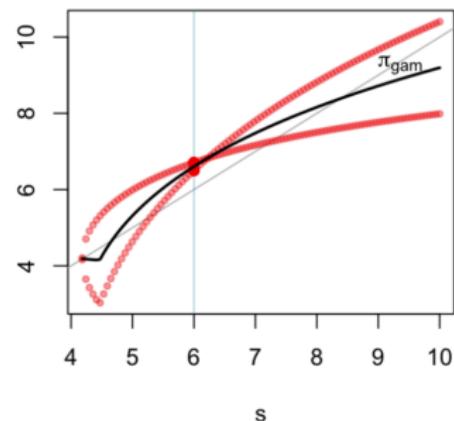
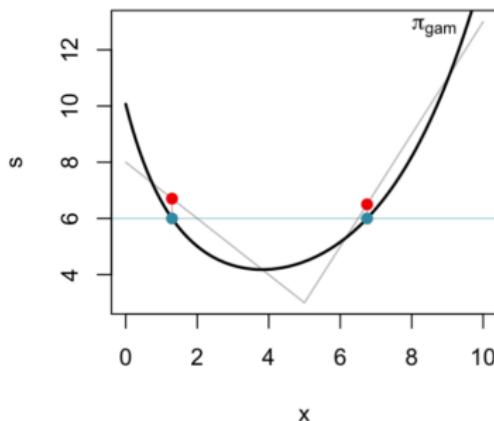


$$s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s] = \mu(\hat{\pi}^{-1}(s)) \text{ since } \mu(x) = \mathbb{E}[Y|X = x].$$

Computing $\mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = s]$

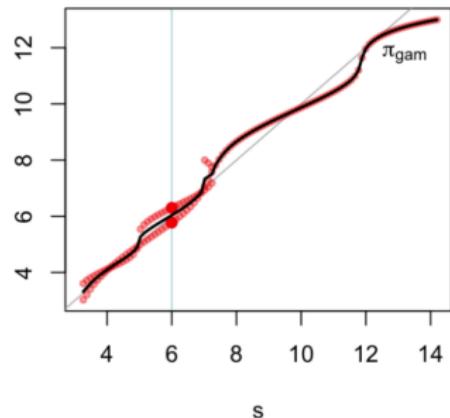
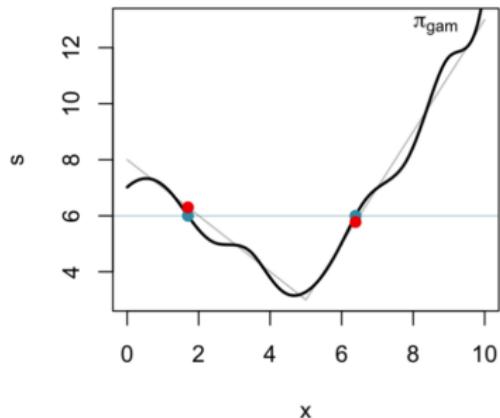
True model $\mu(x)$ and GAM model $\hat{\pi}(x)$

- ▶ select s , e.g. $s = 6$
- ▶ compute $\mathcal{X}_s^{\hat{\pi}} = \{\mathbf{x} \in \mathcal{X}, \hat{\pi}(\mathbf{x}) = s\}$, here $\{1.6; 6.7\}$
- ▶ compute $\mu(\hat{\pi}^{-1}(s))$, $\{6.4, 6.2\}$ and its mean $\bar{\mu}(\hat{\pi}^{-1}(s))$, 6.3
- ▶ plot $(s, \bar{\mu}(\hat{\pi}^{-1}(s)))$



Computing $\mathbb{E}[Y|\hat{\pi}(X) = s]$

True model $\mu(x)$ and GAM model $\hat{\pi}(x)$ (more degrees of freedom)
Plot $s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s]$, should be close to the first diagonal



Seems locally unbiased...

- ▶ impossible to get the figure on the left in higher dimension
- ▶ we used here μ but in practice, μ is unknown !

$\mathbb{E}[Y|\hat{\pi}(X) = s]$: empirical version

How to plot $s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s]$?

but in real life, μ is unknown

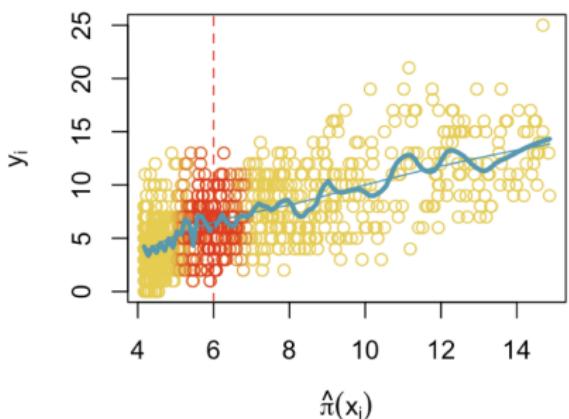
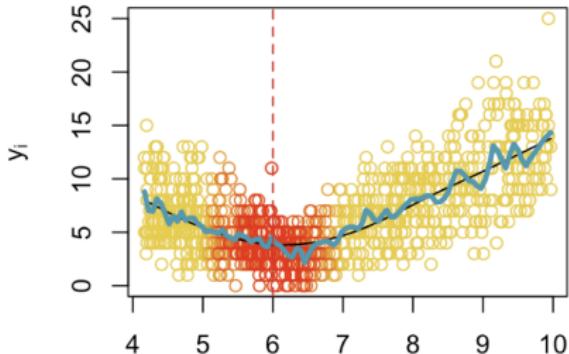
Consider $\{(\hat{\pi}(x_i), y_i)\}$

and fit a local regression

- ▶ fit a model $\hat{\pi}$
- ▶ estimate $\mathbb{E}[Y|\hat{\pi}(X) = s]$
local regression on $\{(\hat{\pi}(x_i), y_i)\}$
- ▶ local (multiplicative) correction

$$\lambda_\alpha(s) = \frac{\mathbb{E}[Y|\hat{\pi}(X) = s]}{s}$$

- ▶ correct $\hat{\pi}$
- $$\hat{\pi}_{BC}(x) = \lambda_\alpha(\hat{\pi}(x)) \cdot \hat{\pi}(x)$$

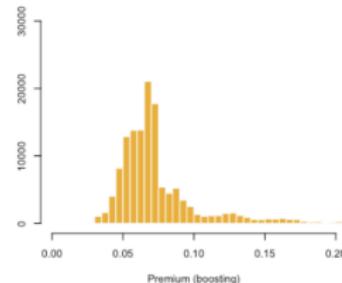
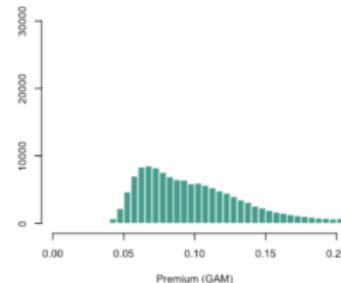
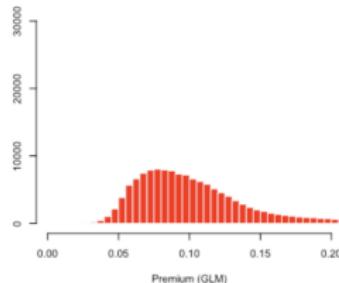


Application of a motor-insurance dataset

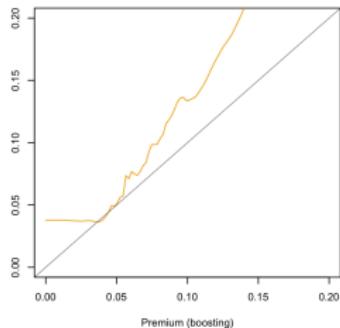
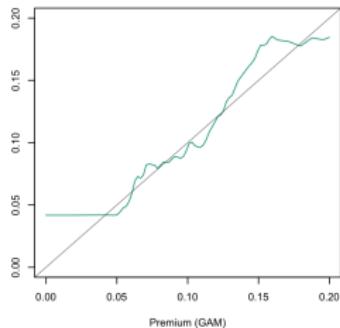
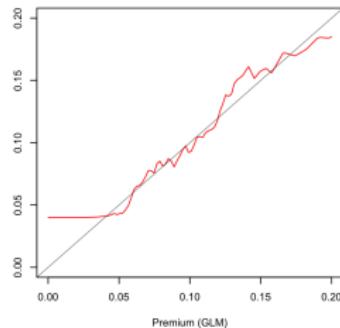
Here we focus only on claims (annual) frequency, corrected from the exposure, `freMTPL2freq` from `CASDataset` package

	π^{glm}	π^{gam}	π^{bst}
average $\bar{\pi}$	0.1087	0.1092	0.0820
10% quantile	0.0605	0.0598	0.0498
90% quantile	0.1682	0.1713	0.1244

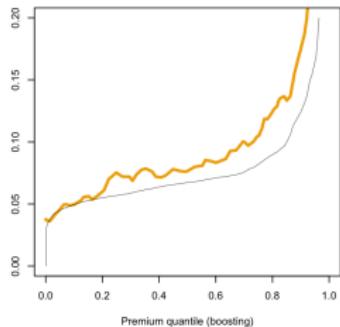
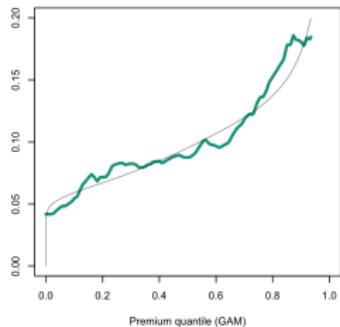
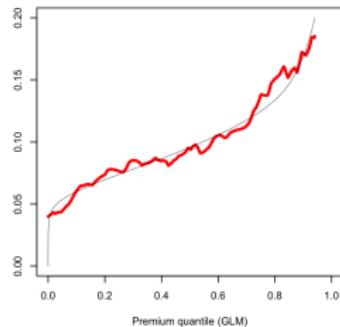
Table 1: Summary statistics on $\{\pi(\mathbf{x}_1), \dots, \pi(\mathbf{x}_n)\}$, on the validation dataset (assuming an exposure of 1 to provide annualized predictions).



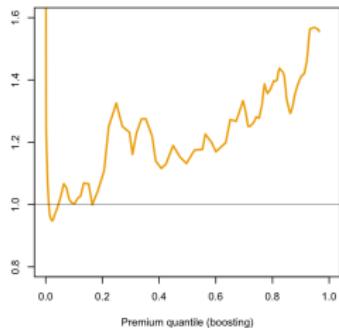
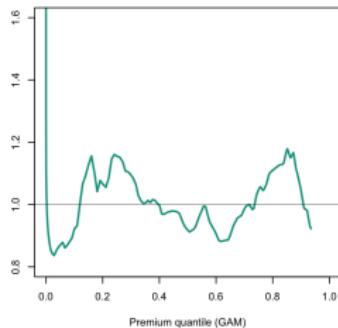
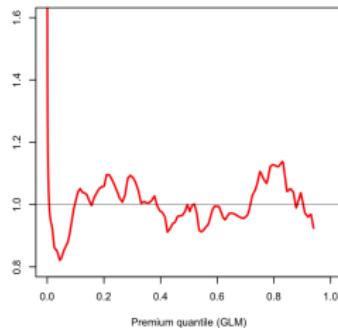
Application of a motor-insurance dataset



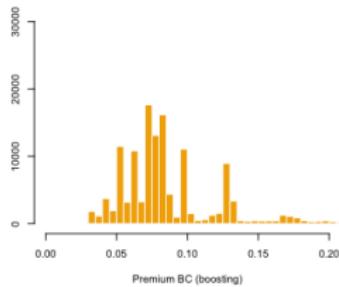
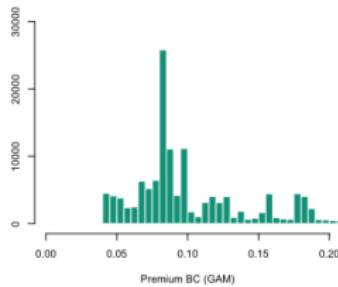
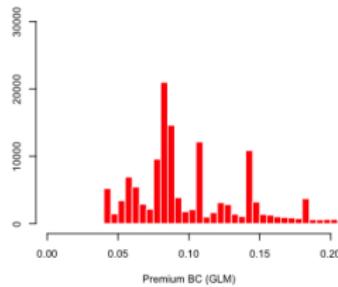
Evolution of $s \mapsto \mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = s]$ and $u \mapsto \mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = F_\pi^{-1}(u)]$



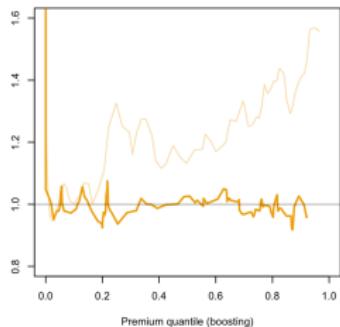
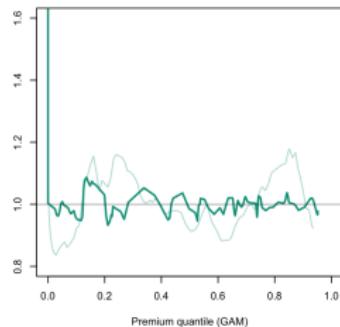
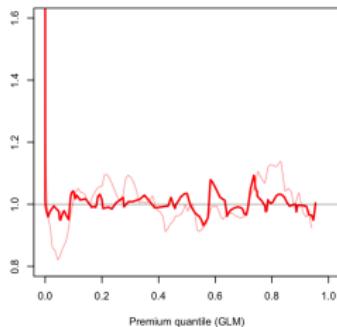
Application of a motor-insurance dataset



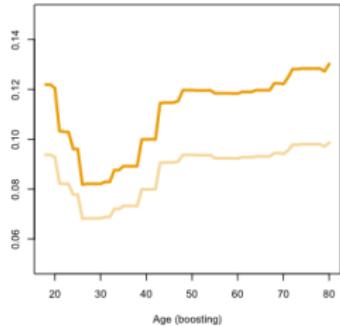
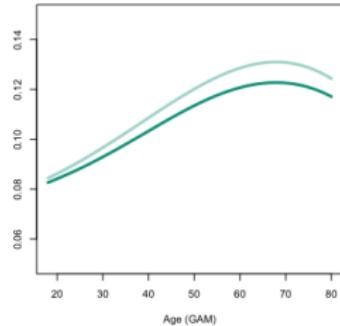
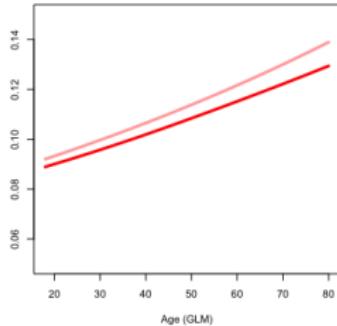
Multiplicative correction $\lambda_\alpha(u) = \mathbb{E}[Y|\hat{\pi}(\mathbf{X}) = F_{\hat{\pi}}^{-1}(u)]/F_{\hat{\pi}}^{-1}(u)$



Application of a motor-insurance dataset



λ_α on the corrected model $\hat{\pi}_{BC}$, and some partial dependence plot
(on the age of the driver)



Wrap-Up

- ▶ fit a model $\hat{\pi}$ using an ML algorithm
- ▶ estimate $\mathbb{E}[Y|\hat{\pi}(X)]$ with local regression on $\{(\hat{\pi}(\mathbf{x}_i), y_i)\}$
- ▶ local (multiplicative) correction $\lambda_\alpha(s) = \frac{\mathbb{E}[Y|\hat{\pi}(X) = s]}{s}$
- ▶ correct $\hat{\pi}$ by setting $\hat{\pi}_{BC}(\mathbf{x}) = \lambda_\alpha(\hat{\pi}(\mathbf{x})) \cdot \hat{\pi}(\mathbf{x})$

Denuit, M., Charpentier, A. & Trufin, J. (2021). [Autocalibration and Tweedie-dominance for Insurance Pricing with Machine Learning](#), ArXiv:2103.03635

Market simulation competition : Completed

Bonus round: Ageless dataset: Completed

#supervised_learning

#educational

#insurance



Motor insurance market simulation



By Imperial CPG

59k

1943

215

10.5k

88

Follow

Overview

Leaderboard

Notebooks

Resources

Submissions

Pick submission for profit leaderboard

Rules

More ▾

Participate

✓ Overview

✗ Cheapest-wins
Market

★ Leaderboards

⊕ Evaluation Metric

camera Market Rules

📊 Weekly Market
Feedback

💾 Dataset

✉️ How To Submit

💡 Announcing our community engagement prizes!

👉 Join the office hours on Discord! (Wednesday 2PM CET)

❓ Have a question? Visit the discussion forum

🐍 Python Starter Notebook

🇷 R Starter Notebook

</> Code based starter kit

CHAT

115 ONLINE

✓ Overview

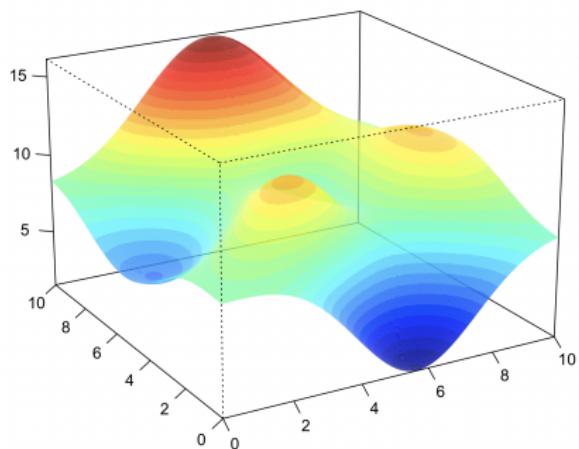
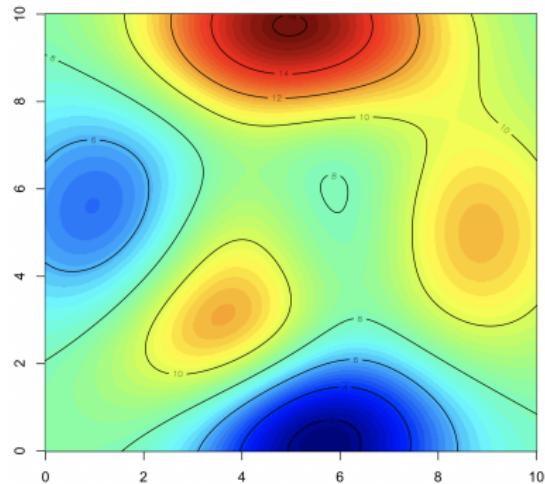
PARTICIPANTS



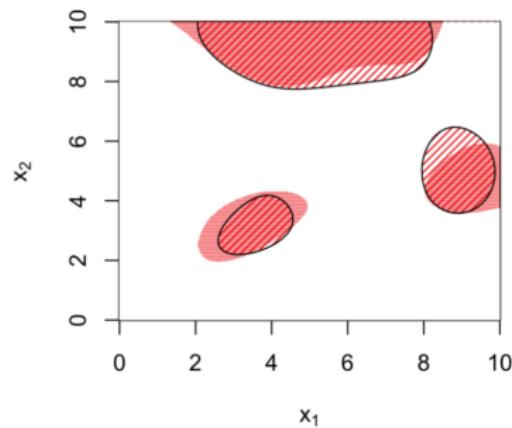
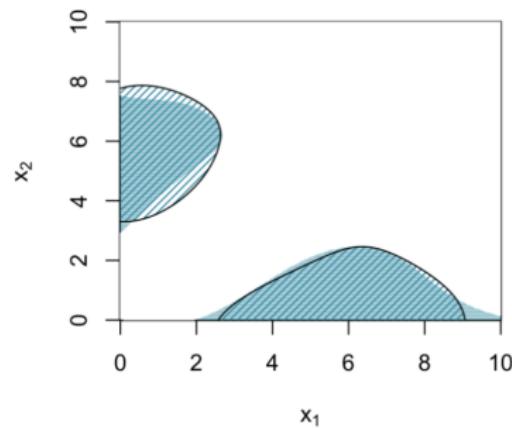
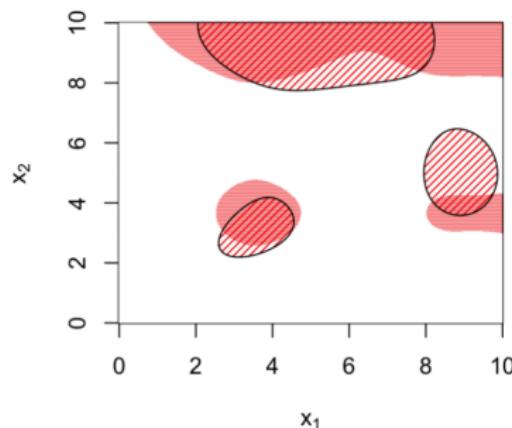
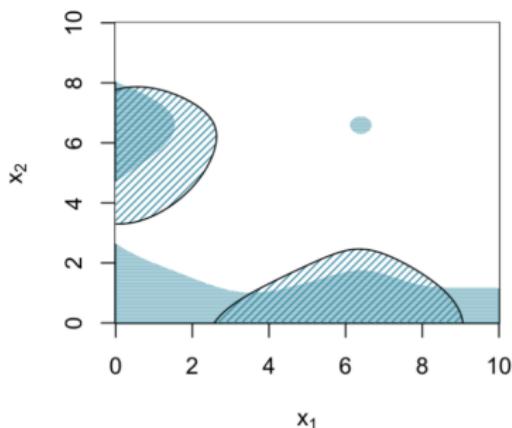
GETTING STARTED

Appendix: Bivariate simulated data

Generate some data $(x_{1,i}, x_{2,i}, y_i)$ where $Y|\mathbf{X}$ has some distribution with mean $\mathbb{E}[Y|\mathbf{x}] = \mu(\mathbf{x})$



Model inversion, $\mathcal{X}_s^{\hat{\pi}} = \{x \in \mathcal{X}, \hat{\pi}(x) \leq s\}$



$\mathcal{X}_s^{\widehat{\pi}}$, $Y | \mathbf{X} \in \mathcal{X}_s^{\widehat{\pi}}$ and $\mathbb{E}[Y | \widehat{\pi}(\mathbf{X}) = s]$: $s = 10$

