

# Missing data, IPW, Imputation, Covariate shift

David S. Rosenberg

NYU: CDS

January 23, 2021

# Contents

- 1 Missing data setup
- 2 Inverse propensity score estimators
- 3 Variance of IPW
- 4 Self-normalized IPW for MAR
- 5 Regression imputation
- 6 Covariate shift
- 7 Importance-weighted regression imputation

## Missing data setup

---

# MCAR setup

- We want to estimate  $\mathbb{E}Y$  for  $Y \sim p(y)$ .
- **Full data:**  $Y_1, \dots, Y_n$  i.i.d.  $p(y)$ .
- Response indicators:  $R, R_1, \dots, R_n \in \{0, 1\}$  are i.i.d.
  - with  $\mathbb{P}(R = 1) = \pi$ .
- What we actually observe:

$$(R_1, R_1 Y_1), \dots, (R_n, R_n Y_n).$$

- **Complete cases** are observations with  $R_i = 1$ .
- **Incomplete cases** are observations with  $R_i = 0$ .
- **MCAR assumption:**  $R_i \perp\!\!\!\perp Y_i$  for each  $i$

# MAR setup

- Assume we have **covariate**  $X_i$  about each individual  $i$ .
- Also assume that  $X_i$  is **never missing**.
- Full data:  $(X_1, Y_1), \dots, (X_n, Y_n)$  i.i.d  $p(x, y)$ .
- What we actually observe:

$$(X_1, R_1, R_1 Y_1), \dots, (X_n, R_n, R_n Y_n).$$

- **MAR assumption:**  $R_i \perp\!\!\!\perp Y_i \mid X_i$  for each  $i$ 
  - i.e.  $p(r, y \mid x) = p(r \mid x)p(y \mid x)$

# The propensity score

- Key piece in the MAR setting is the model for missingness:

$$\mathbb{P}(R = 1 \mid X = x, Y = y) = \mathbb{P}(R = 1 \mid X = x) = \pi(x).$$

- $\pi(x)$  is called the **propensity score**.
- If the propensity score is 0, we have a blind spot in our input space
  - can't do anything about it (at least with our estimators)

## Assumption

Unless otherwise noted, we will always assume that propensity scores are strictly positive:  
 $\pi(x) > 0$ .

## Inverse propensity score estimators

# Observed responses represent multiple unobserved responses

- Suppose  $\mathcal{X} = \{0, 1\}$  corresponding to two types of people.
- When  $X_i = 1$ , response probability  $\pi(1) = \mathbb{P}(R_i = 1 \mid X_i = 1) = 0.2$
- When  $X_i = 0$ , response probability  $\pi(0) = \mathbb{P}(R_i = 1 \mid X_i = 0) = 0.1$
- If we observe  $Y_i$  for an individual with  $X_i = 1$ ,
  - That individual represents roughly 5 people from the full data.
- If we observe  $Y_i$  for an individual with  $X_i = 0$ ,
  - That individual represents roughly 10 people from the full data.



# Inverse propensity weighted (IPW) Mean

- When<sup>1</sup>  $\pi(x) > 0 \forall x \in \mathcal{X}$ ,  
we can define the **IPW mean estimator** for  $\mathbb{E}Y$ :

$$\begin{aligned}\hat{\mu}_{\text{ipw}} &= \frac{1}{n} \sum_{i: R_i=1} \frac{Y_i}{\pi(X_i)} \\ &= \frac{1}{n} \sum_{i=1}^n \frac{R_i Y_i}{\pi(X_i)}\end{aligned}$$

- $\hat{\mu}_{\text{ipw}}$  is **unbiased** for  $\mathbb{E}Y$  (i.e.  $\mathbb{E}\hat{\mu}_{\text{ipw}} = \mathbb{E}Y$ .)
- $\hat{\mu}_{\text{ipw}}$  is **consistent** for  $\mathbb{E}Y$  (i.e.  $\hat{\mu}_{\text{ipw}} \xrightarrow{P} \mathbb{E}Y$  as  $n \rightarrow \infty$ )

---

<sup>1</sup>We assume here and everywhere below that  $\pi(x) > 0 \forall x \in \mathcal{X}$ .

# The complete case mean estimator

- The **complete case mean estimator** is defined as

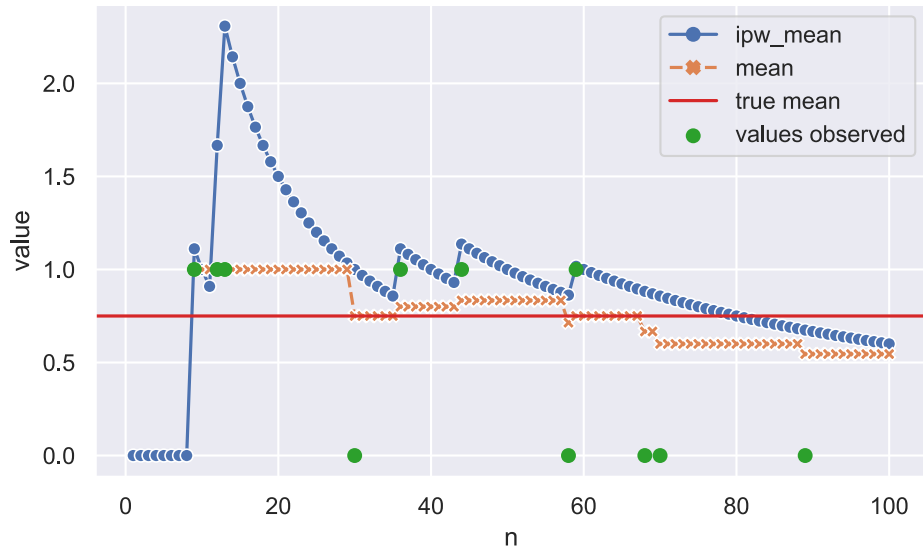
$$\hat{\mu}_{\text{cc}} = \frac{\sum_{i=1}^n R_i Y_i}{\sum_{i=1}^n R_i}.$$

- i.e. just take the average of the observed  $Y_i$ 's

# MCAR example

- For  $i = 1, \dots, n$ 
  - Let  $R_i$  be independent of  $(X_i, Y_i)$ .
  - $\pi(x) = \mathbb{P}(R_i = 1 \mid X_i = x) \equiv 0.1$ .
  - Let  $Y_i \in \{0, 1\}$  with  $\mathbb{P}(Y_i = 1) = 0.75$

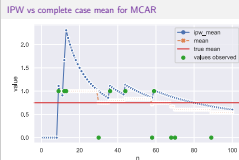
## IPW vs complete case mean for MCAR



## DS-GA 3001: Tools and Techniques for ML

## └ Inverse propensity score estimators

## └ IPW vs complete case mean for MCAR



- We've added in the orange line here, which represent the mean of the complete cases.
- Note that the orange line is constant between observations, and jumps whenever we get a new observation.

## IPW vs complete case mean, 5000x

- We repeat the experiment<sup>2</sup> above 5000 times ( $n = 100$  samples in each) and get the following.
- Recall that the true mean is  $\mu = 0.75$ .

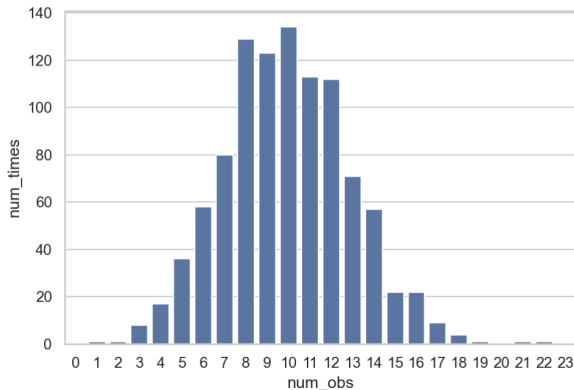
estimator	mean	SD	SE	bias	RMSE
mean	0.751654	0.143943	0.002036	0.001654	0.143953
ipw_mean	0.752540	0.262224	0.003708	0.002540	0.262237

<sup>2</sup>In the very rare event that the sample in a particular repetition has 0 complete cases, we take  $\hat{\mu}_{cc} = 0$ .

## Variance of IPW

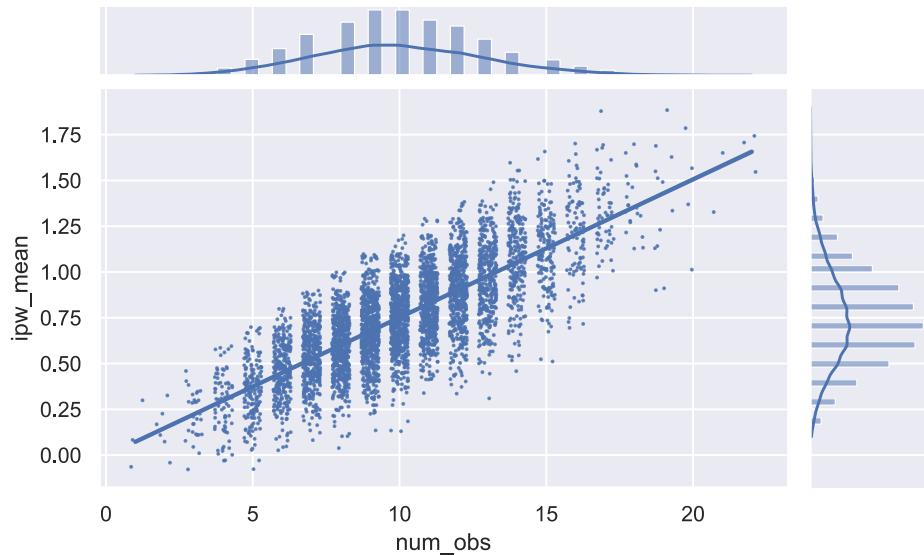
## Probability review: how many responses will we get?

- $R, R_1, \dots, R_n \in \{0, 1\}$  are i.i.d. with  $\mathbb{P}(R = 1) = 0.1$ .
- Number of observations:  $N = \sum_{i=1}^n R_i$ .
- Expected number of responses is  $\mathbb{E}N = 0.1n$  and  $N \sim \text{Binom}(n, p = 0.1)$ .
- Histogram of  $N$  from 1000 simulations of our setup with  $n = 100$ :

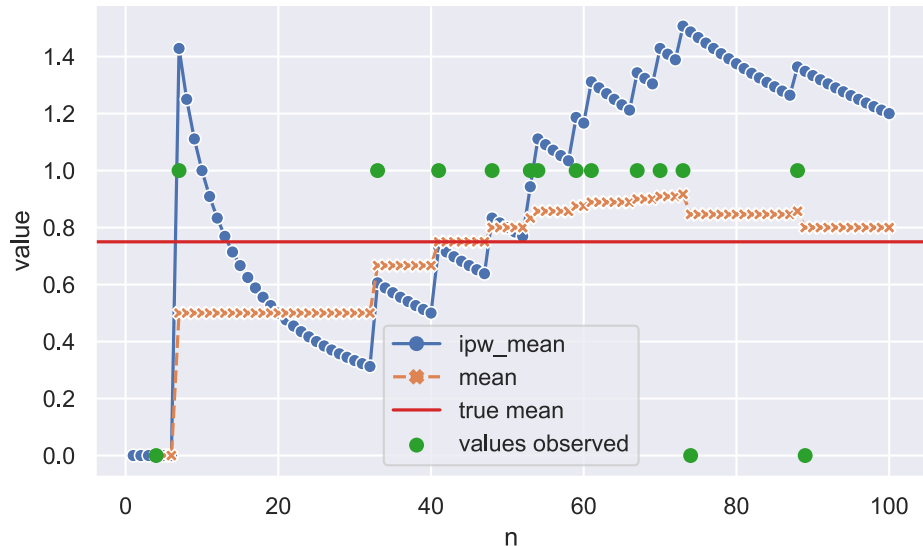




## IPW mean vs number of observations



## IPW mean for “too many” observations



## IPW: Can we improve this estimator for MCAR?

- IPW for constant observation probability  $\pi$ :

$$\hat{\mu}_{\text{ipw}} = \frac{1}{n} \sum_{i=1}^n \frac{R_i Y_i}{\pi}$$

- Idea: Rather than dividing by  $n$ ,
  - divide by actual number of observations  $N = \sum_{i=1}^n R_i$ .
- This exactly gives us back

$$\hat{\mu}_{\text{cc}} = \frac{\sum_{i=1}^n R_i Y_i}{\sum_{i=1}^n R_i}.$$

- Have we gone in a useless circle?
- Not at all! Let's try to apply this "correction" to the more general MAR case. . .

## Self-normalized IPW for MAR

- We can write

$$\hat{\mu}_{\text{ipw}} = \frac{1}{n} \sum_{i=1}^n \frac{R_i Y_i}{\pi(X_i)} = \frac{1}{n} \sum_{i=1}^n W_i R_i Y_i,$$

where

$$W_i := \frac{1}{\pi(X_i)} = \frac{1}{p(R_i = 1 \mid X_i)}.$$

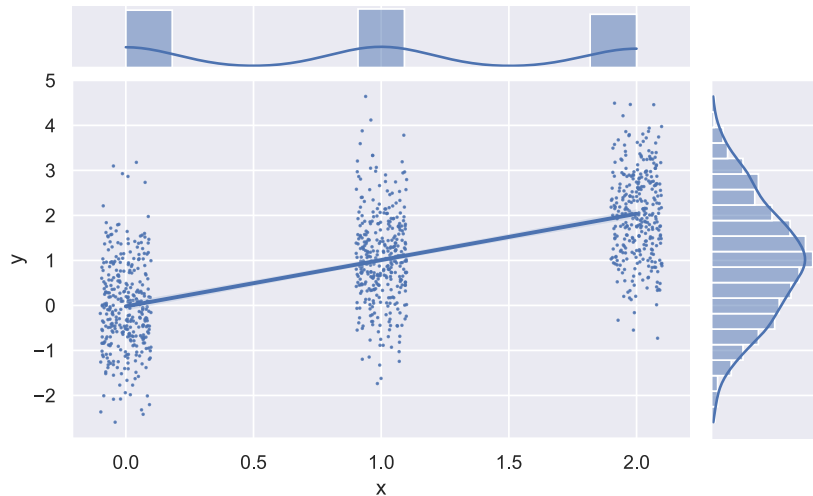
- We'll refer to  $W_i$  as the **weight** for observation  $Y_i$ .
- It's like each observed response  $Y_i$  (with  $R_i = 1$ ) represents  $W_i$  responses in the full data.
- Upweighting by  $W_i$  makes up for the zeros when  $R_i = 0$ .

## IPW in MAR: very large or small number of observations?

- If each observed response  $Y_i$  represents  $W_i$  responses in the full data,
  - then our observed data represents  $\sum_{i=1}^n W_i R_i$  people.
- The IPW estimate normalizes by  $n$ :  $\hat{\mu}_{\text{ipw}} = \frac{1}{n} \sum_{i=1}^n W_i R_i Y_i$ .
- But what if  $\sum_{i=1}^n W_i R_i$  is much smaller or larger than  $n$ ?
- Then it seems like we're normalizing by the wrong thing...

## MAR: “SeaVan1” distribution illustrated

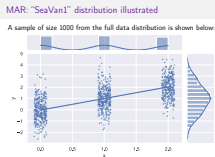
A sample of size 1000 from the full data distribution is shown below:



## DS-GA 3001: Tools and Techniques for ML

## └ Self-normalized IPW for MAR

## └ MAR: “SeaVan1” distribution illustrated

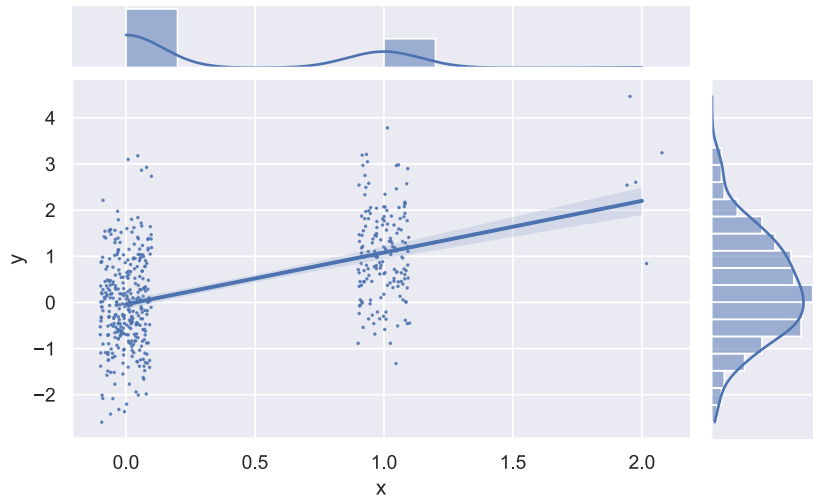


We've added jitter to the x values so that it's easier to see the distribution.



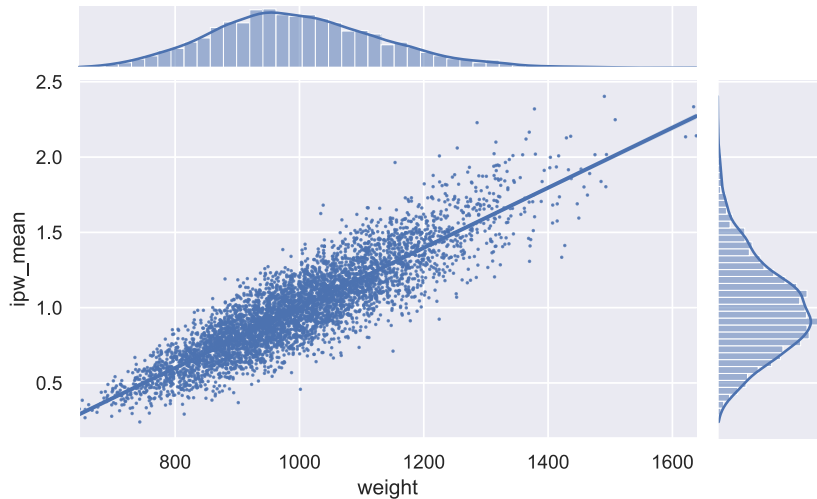
## MAR: “SeaVan1” distribution illustrated

$(X_i, Y_i)$  for which  $R_i = 1$ , i.e. the complete cases.



# IPW vs total weight of observations

- The points below have correlation 0.885.



# The self-normalized IPW estimator

- If we normalize by  $\sum_{i=1}^n W_i R_i$  instead of  $n$ , we get

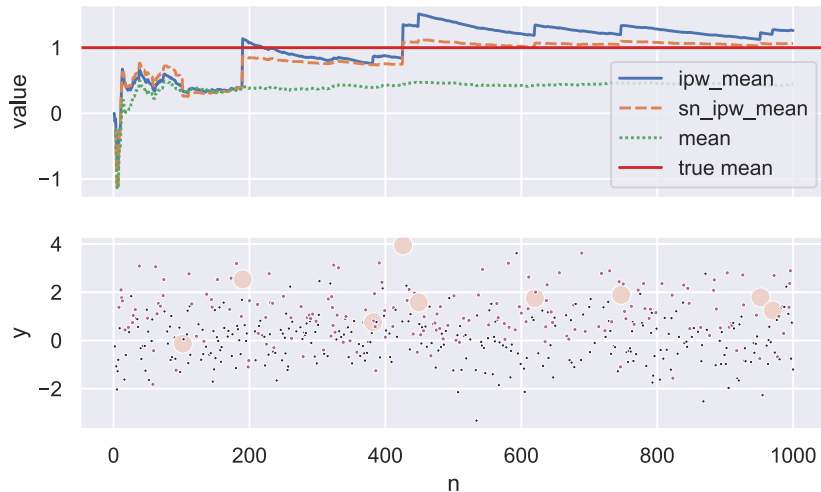
## Definition (Self-normalized IPW mean)

For a dataset  $(W_1, R_1, Y_1), \dots, (W_n, R_n, Y_n)$  as described above,

$$\hat{\mu}_{\text{sn\_ipw}} = \frac{\sum_{i=1}^n W_i R_i Y_i}{\sum_{i=1}^n W_i R_i}$$

- In the MCAR case with  $\pi(x) \equiv p$ ,  $\hat{\mu}_{\text{sn\_ipw}} = \hat{\mu}_{\text{cc}}$  and seems preferable to  $\hat{\mu}_{\text{ipw}}$ .

# Self-normalized IPW estimator on SeaVan1



## IPW vs self-normalized IPW: 5000x

- We repeat the experiment above 5000 times (1000 samples each) and get the following.
- Recall that the true mean is  $\mu = 1.0$ .

estimator	mean	SD	SE	bias	RMSE
mean ( $\hat{\mu}_{cc}$ )	0.357244	0.050305	0.000711	-0.643534	0.645497
ipw_mean ( $\hat{\mu}_{ipw}$ )	0.995142	0.308634	0.004365	-0.005635	0.308686
sn_ipw_mean ( $\hat{\mu}_{sn\_ipw}$ )	0.978119	0.197319	0.002791	-0.022659	0.198615

## Regression imputation

---

## Regression imputation: basic idea

$X$	$R$	$Y$
$x_1$	1	$y_1$
$x_2$	0	?
$x_3$	0	?
$x_4$	1	$y_4$
$\vdots$	$\vdots$	$\vdots$
$x_n$	1	$y_n$

 $\Rightarrow$ 

$X$	$R$	$Y$
$x_1$	1	$y_1$
$x_2$	0	$\hat{f}(x_2)$
$x_3$	0	$\hat{f}(x_3)$
$x_4$	1	$y_4$
$\vdots$	$\vdots$	$\vdots$
$x_n$	1	$y_n$

- Fit  $\hat{f}(x)$  on **complete cases** ( $R = 1$ ) to approximate  $\mathbb{E}[Y \mid X = x]$ .
- **Regression imputation estimator:** Estimate  $\mathbb{E}Y$  with

$$\frac{1}{n} \left( y_1 + \hat{f}(x_2) + \hat{f}(x_3) + y_4 + \cdots + y_n \right).$$

# Well-specified and misspecified models

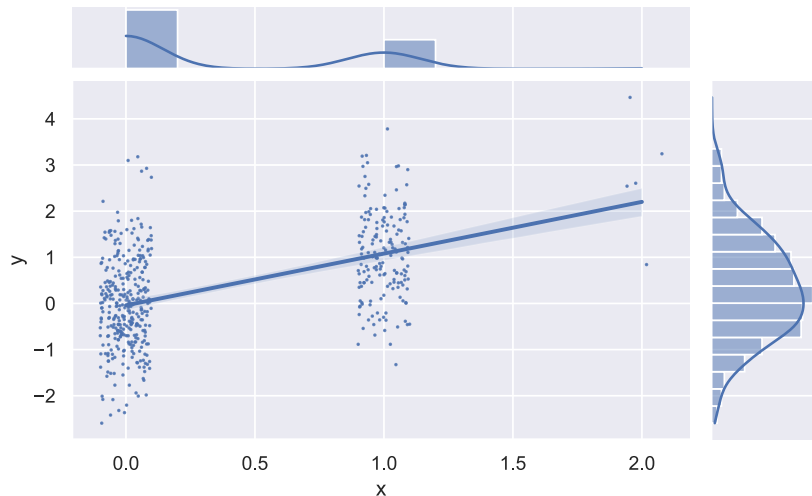
- In statistics, a **model** is a set of distributions
  - (or conditional distributions).
- A model is **well specified** if it contains the data-generating distribution.
  - Also referred to as **correctly specified**.
- If a model is not well specified, we say it's **misspecified** or **incorrectly specified**.
- We'll see that regression imputation has the following performance characteristics:

	MCAR	MAR
well specified	Good	Good
misspecified	OK/Good	Bad



# MAR: SeaVan1 distribution illustrated

$(X_i, Y_i)$  for which  $R_i = 1$ , i.e. the complete cases.



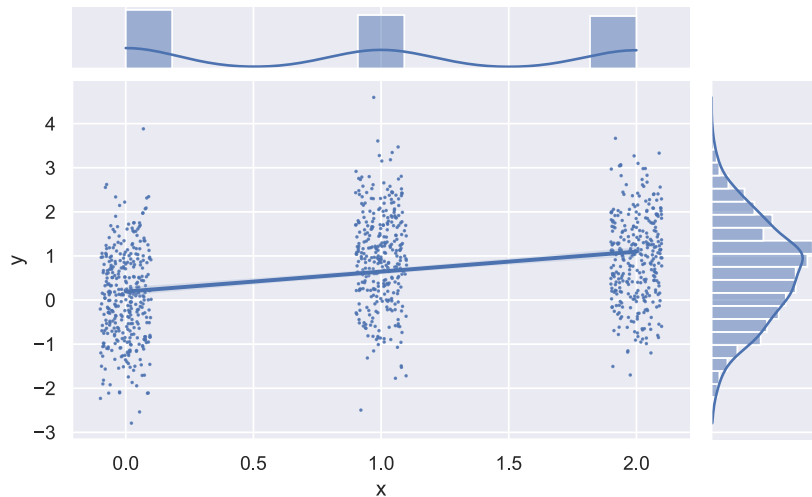
## Performance on SeaVan1

- Fit  $\hat{f}(x) = a + bx$  to the complete cases.
- Impute missing  $Y_i$ 's with  $\hat{f}(X_i)$ ...

estimator	mean	SD	SE	bias	RMSE
mean ( $\hat{\mu}_{cc}$ )	0.3572	0.0503	0.0007	-0.6435	0.6455
ipw_mean ( $\hat{\mu}_{ipw}$ )	0.9951	0.3086	0.0044	-0.0056	0.3087
sn_ipw_mean ( $\hat{\mu}_{sn\_ipw}$ )	0.9781	0.1973	0.0028	-0.0227	0.1986
impute_linear ( $\hat{\mu}_{\hat{f}}$ )	0.9989	0.0777	0.0011	-0.0018	<b>0.0777</b>

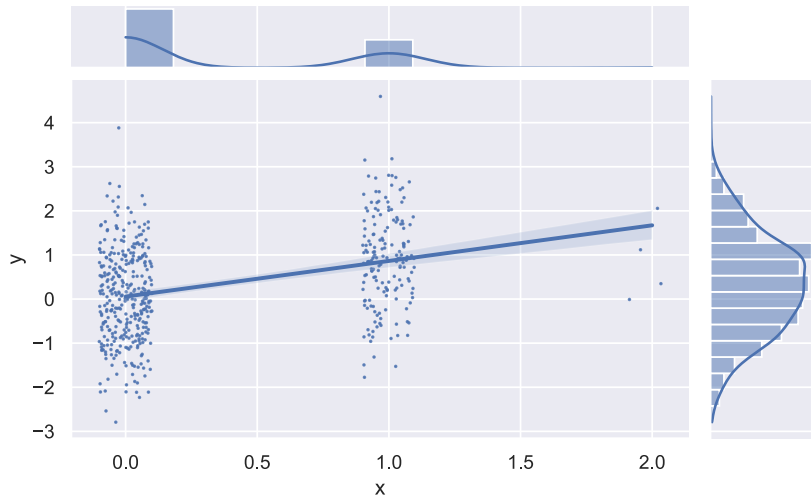
## MAR: “SeaVan2” distribution illustrated

- Full data for sample of size  $n = 1000$ ;  $\mathbb{E}[Y | X = x] = \mathbb{1}[x \geq 1]$ .



## MAR: “SeaVan2” distribution illustrated

- Complete cases in sample of size  $n = 1000$



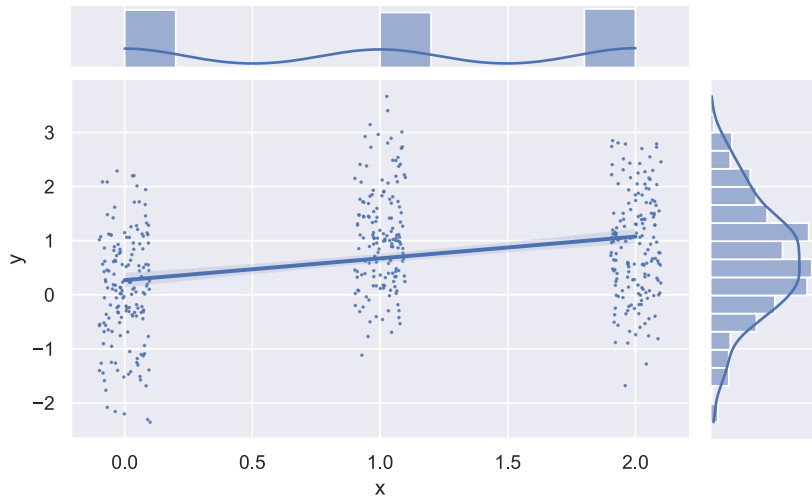
## Performance on SeaVan2

- Fit  $\hat{f}(x) = a + bx$  to the complete cases.

estimator	mean	SD	SE	bias	RMSE
mean ( $\hat{\mu}_{cc}$ )	0.3453	0.0497	0.0007	-0.3221	0.3259
ipw_mean ( $\hat{\mu}_{ipw}$ )	0.6634	0.1977	0.0028	-0.0040	0.1978
sn_ipw_mean ( $\hat{\mu}_{sn\_ipw}$ )	0.6580	0.1462	0.0021	-0.0094	0.1465
impute_linear ( $\hat{\mu}_{\hat{f}}$ )	0.9382	0.0793	0.0011	0.2708	<b>0.2821</b>

## SeaVan2\_MCAR illustrated

- Complete cases in sample size  $n = 1000$



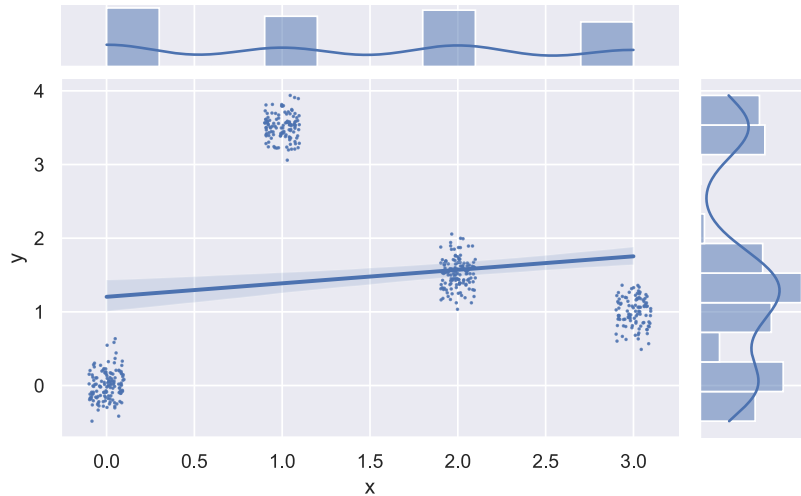
## Performance on SeaVan2\_MCAR

- Fit  $\hat{f}(x) = a + bx$  to the complete cases.
- True mean: 0.667

estimator	mean	SD	SE	bias	RMSE
mean ( $\hat{\mu}_{cc}$ )	0.66724	0.05059	0.00226	0.00116	0.05061
ipw_mean ( $\hat{\mu}_{ipw}$ )	0.66712	0.05552	0.00248	0.00104	0.05553
sn_ipw_mean ( $\hat{\mu}_{sn\_ipw}$ )	0.66724	0.05059	0.00226	0.00116	0.05061
impute_linear ( $\hat{\mu}_{\hat{f}}$ )	0.66763	0.04953	0.00222	0.00155	<b>0.04955</b>

## MCAR\_normal\_nonlinear

Complete cases for  $\mathbb{P}(R = 1 \mid X) \equiv 0.5$  and  $n = 1000$ :





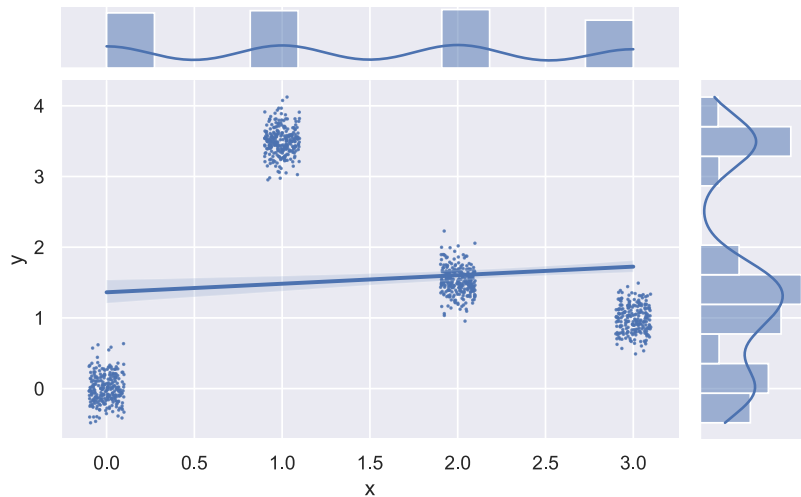
## Performance on MCAR\_normal\_nonlinear

- True mean: 1.50

estimator	mean	SD	SE	bias	RMSE
mean	1.5021	0.0593	0.0019	0.0009	0.0593
ipw_mean	1.5014	0.0759	0.0024	0.0002	0.0759
sn_ipw_mean	1.5021	0.0593	0.0019	0.0009	0.0593
impute_linear	1.5030	0.0592	0.0019	0.0018	<b>0.0592</b>

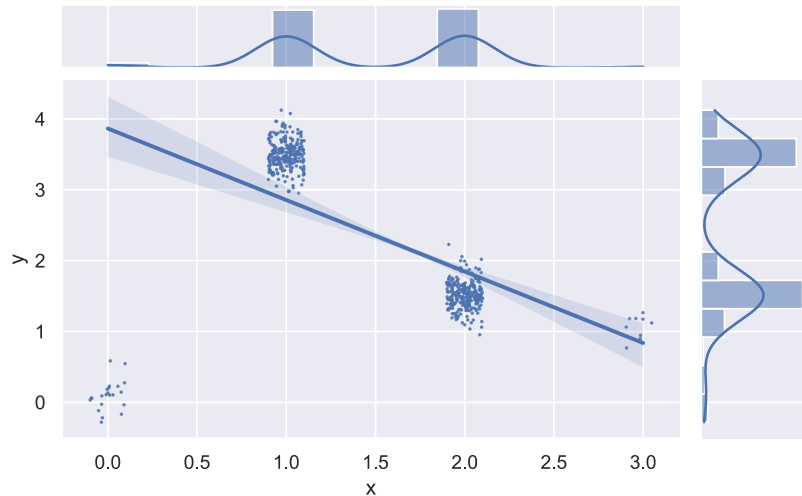
## MAR\_normal\_nonlinear

Full data for  $n = 1000$ :



## MAR\_normal\_nonlinear

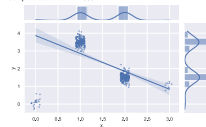
Complete cases for  $n = 1000$ :



## └ Regression imputation

## └ MAR\_normal\_nonlinear

MAR\_normal\_nonlinear

Complete cases for  $n = 1000$ .

Note that the linear fit is completely off from the fit to the full data (preceding slide) because of the sample bias.

## Performance on MAR\_normal\_nonlinear

- True mean: 1.50

estimator	mean	SD	SE	bias	RMSE
mean	2.4075	0.0476	0.0015	0.9063	0.9075
ipw_mean	1.4985	0.0851	0.0027	-0.0027	0.0852
sn_ipw_mean	1.5070	0.1224	0.0039	0.0057	0.1225
impute_linear	2.4060	0.0583	0.0018	0.9048	<b>0.9066</b>

# What's going on?

- The best linear fit to the complete cases is
  - COMPLETELY DIFFERENT from the best linear fit to full data.
- Essential issue: model is fit to the **complete cases**,
  - but applied on **incomplete cases**.
- Complete cases and incomplete cases have different distributions!

## Covariate shift

# Supervised learning framework

- $\mathcal{X}$ : input space
- $\mathcal{Y}$ : outcome space
- $\mathcal{A}$ : action space
- **Prediction function**  $f : \mathcal{X} \rightarrow \mathcal{A}$  (takes input  $x \in \mathcal{X}$  and produces action  $a \in \mathcal{A}$ )
- **Loss function**  $\ell : \mathcal{A} \times \mathcal{Y} \rightarrow \mathbb{R}$  (evaluates action  $a$  in the context of outcome  $y$ ).



- Let  $(X, Y) \sim p(x, y)$ .
- The **risk** of a prediction function  $f : \mathcal{X} \rightarrow \mathcal{A}$  is  $R(f) = \mathbb{E}\ell(f(X), Y)$ .
  - the expected loss of  $f$  on a new example  $(X, Y) \sim p(x, y)$
- Ideally we'd find the **Bayes prediction function**  $f^* \in \arg \min_f R(f)$ .

# Empirical risk minimization

- Training data:  $\mathcal{D}_n = ((X_1, Y_1), \dots, (X_n, Y_n))$ 
  - drawn i.i.d. from  $p(x, y)$ .
- Let  $\mathcal{F}$  be a **hypothesis space** of functions mapping  $\mathcal{X} \rightarrow \mathcal{A}$
- A function  $\hat{f}$  is an **empirical risk minimizer** over  $\mathcal{F}$  if

$$\hat{f} \in \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \ell(f(X_i), Y_i).$$

- Uses sample  $\mathcal{D}_n$  from  $p(x, y)$  to estimate expectation w.r.t.  $p(x, y)$ .
- Most machine learning methods can be written in this form.
- What if we only have a sample from another distribution  $q(x, y)$ ?

## Covariate shift

- Goal: Find  $f$  minimizing risk  $R(f) = \mathbb{E}\ell(f(X), Y)$  where

$$(X, Y) \sim p(x, y) = p(x)p(y | x).$$

- Standard:  $\mathcal{D}_n = ((X_1, Y_1), \dots, (X_n, Y_n))$  is i.i.d. from

$$p(x, y) = p(x)p(y | x).$$

- **Covariate shift:**  $\mathcal{D}_n = ((X_1, Y_1), \dots, (X_n, Y_n))$  is i.i.d. from

$$q(x, y) = q(x)p(y | x).$$

- The covariate distribution has changed, but
  - the conditional distribution  $p(y | x)$  is the same in both cases.

## Covariate shift: the issue

- Under covariate shift,

$$\mathbb{E}_{(X_i, Y_i) \sim q(x, y)} \left[ \frac{1}{n} \sum_{i=1}^n \ell(f(X_i), Y_i) \right] \neq \mathbb{E}_{(X, Y) \sim p(x, y)} \ell(f(X), Y).$$

- i.e the empirical risk is a **biased** estimator for risk.
- Naive empirical risk minimization is optimizing the wrong thing.
- Can we get an unbiased estimate of risk with  $\mathcal{D}_n \sim q(x, y)$ ?
- **Importance weighting** is one approach to this problem.

# Importance weighting for covariate shift

- $\mathcal{D}_n = ((X_1, Y_1), \dots, (X_n, Y_n))$  is i.i.d. from

$$q(x, y) = q(x)p(y | x).$$

- Then the **importance-weighted empirical risk** is

$$\begin{aligned}\hat{R}_{IW}(f) &= \frac{1}{n} \sum_{i=1}^n \frac{p(x)p(y | x)}{q(x)p(y | x)} \ell(f(X_i), Y_i) \\ &= \frac{1}{n} \sum_{i=1}^n \frac{p(x)}{q(x)} \ell(f(X_i), Y_i).\end{aligned}$$

- Note that  $\mathbb{E}_{\mathcal{D}_n \sim q(x, y)} \hat{R}_{IW}(f) = \mathbb{E}_{(X, Y) \sim p(x, y)} \ell(f(X), Y)$ .
- So the **importance-weighted empirical risk** is unbiased.

## Potential variance issues

- Since the summands are independent, we have

$$\begin{aligned}\mathrm{Var}\left(\hat{R}_{\mathrm{iw}}(f)\right) &= \mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^n f(X_i)\frac{p(X_i)}{q(X_i)}\right) \\ &= \frac{1}{n}\mathrm{Var}\left(f(X)\frac{p(X)}{q(X)}\right)\end{aligned}$$

- If  $q(x)$  is much smaller than  $p(x)$ ,
  - the importance weight can get very large,
  - variance can blow up.

## Importance-weighted regression imputation

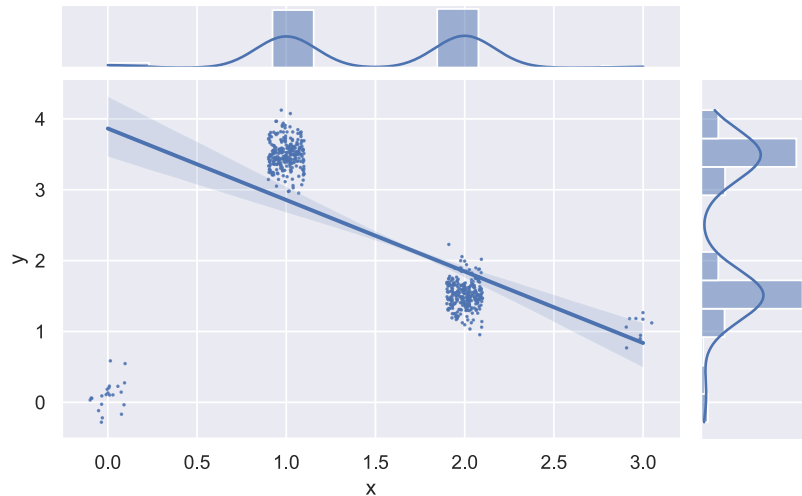
# Importance weighted

- Our linear model is fit to data from the complete case distribution
  - we need it to be fit to the incomplete case distribution
  - or the full data distribution (also common)
- Two new estimators:
  - **impute\_IPW\_linear**: examples weighted by  $\frac{1}{\pi(X_i)}$  so unbiased for full data
  - **impute\_IW\_linear**: examples weighted by  $\frac{1-\pi(X_i)}{\pi(X_i)}$  so unbiased for incomplete data



## Recap: MAR\_normal\_nonlinear

Complete cases for  $n = 1000$ :



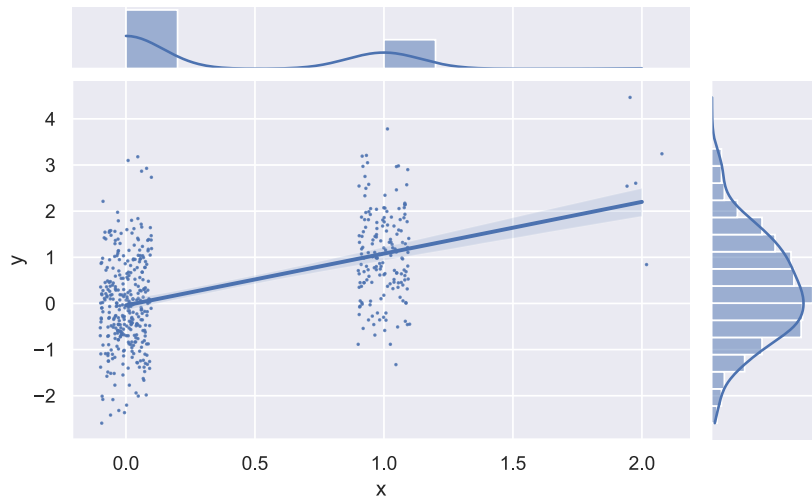
## Performance on MAR\_normal\_nonlinear

- True mean: 1.50

estimator	mean	SD	SE	bias	RMSE
mean	2.4075	0.0476	0.0015	0.9063	0.9075
ipw_mean	1.4985	0.0851	0.0027	-0.0027	0.0852
sn_ipw_mean	1.5070	0.1224	0.0039	0.0057	0.1225
impute_linear	2.4060	0.0583	0.0018	0.9048	<b>0.9066</b>
impute_ipw_linear	1.9895	0.0777	0.0025	0.4883	<b>0.4944</b>
impute_iw_linear	1.5005	0.0466	0.0015	-0.0007	<b>0.0466</b>

## Recap: SeaVan1 distribution illustrated

$(X_i, Y_i)$  for which  $R_i = 1$ , i.e. the complete cases.



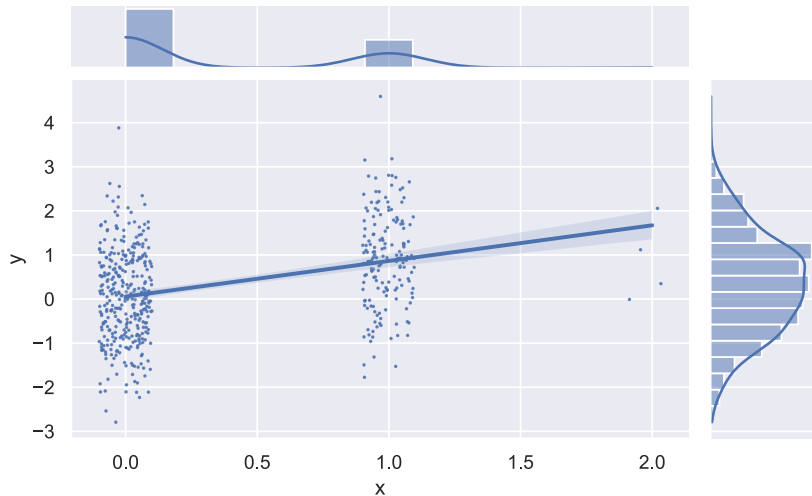
## Performance on SeaVan1

- Fit  $\hat{f}(x) = a + bx$  to the complete cases.

estimator	mean	SD	SE	bias	RMSE
mean	0.3564	0.0515	0.0016	-0.6431	0.6452
ipw_mean	1.0127	0.2968	0.0094	0.0132	0.2971
sn_ipw_mean	0.9906	0.1890	0.0060	-0.0089	0.1892
impute_linear	1.0022	0.0781	0.0025	0.0027	<b>0.0782</b>
impute_ipw_linear	1.0039	0.1439	0.0046	0.0044	<b>0.1440</b>
impute_iw_linear	1.0047	0.1529	0.0048	0.0052	<b>0.1530</b>

## MAR: “SeaVan2” distribution illustrated

- Complete cases in sample of size  $n = 1000$



## Performance on SeaVan2

- Fit  $\hat{f}(x) = a + bx$  to the complete cases.

estimator	mean	SD	SE	bias	RMSE
mean	0.3425	0.0493	0.0007	-0.3244	0.3282
ipw_mean	0.6655	0.1939	0.0027	-0.0014	0.1939
sn_ipw_mean	0.6594	0.1446	0.0020	-0.0075	0.1448
impute_linear	0.9364	0.0792	0.0011	0.2695	<b>0.2809</b>
impute_ipw_linear	0.6750	0.1503	0.0021	0.0081	<b>0.1505</b>
impute_iw_linear	0.6677	0.1561	0.0022	0.0008	<b>0.1561</b>

## Caveat on results

- The importance-weighted regression imputation estimators seem promising.
- The estimators rely on knowing the importance weights  $p(x)/q(x)$ .
- Performance may be significantly worse when we use estimates  $\hat{p}(x)/\hat{q}(x)$ .
- This is something we can explore in homeworks and projects.

## References

---



# References I