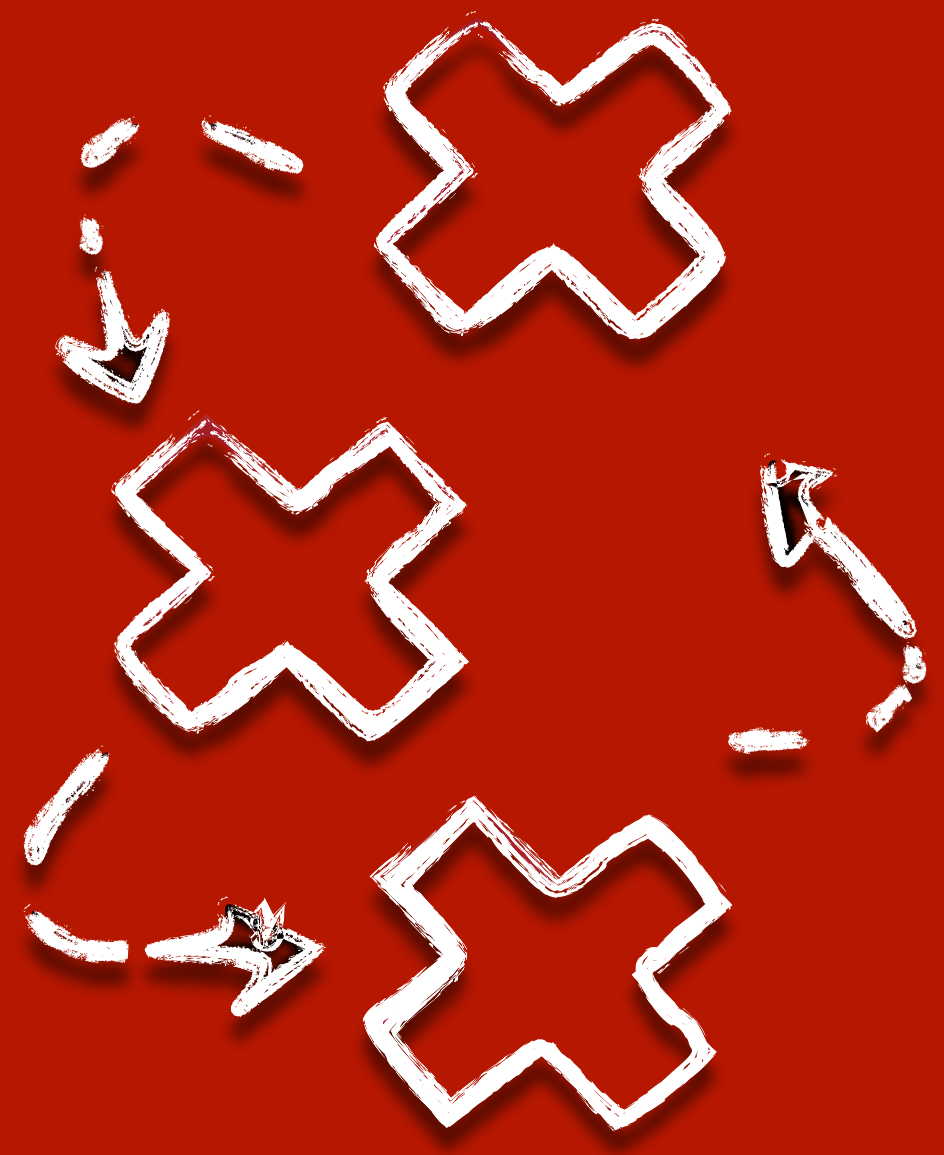


Causal Data Science

Lecture 7:2 Estimating causal effects

Lecturer: Sara Magliacane

UvA - Spring 2024

Estimands for binary treatments

- We generally cannot estimate **unit-level causal effect:** $Y_i(t = 1) - Y_i(t = 0)$
- We can estimate the average causal effect/**average treatment effect**

$$\text{ATE} = \mathbb{E}[Y(t = 1) - Y(t = 0)] = \mathbb{E}[Y \mid \text{do}(T = 1)] - \mathbb{E}[Y \mid \text{do}(T = 0)]$$

Estimands for binary treatments

- We generally cannot estimate **unit-level causal effect:** $Y_i(t=1) - Y_i(t=0)$
- We can estimate the average causal effect/**average treatment effect**

$$\text{ATE} = \mathbb{E}[Y(t=1) - Y(t=0)] = \mathbb{E}[Y \mid do(T=1)] - \mathbb{E}[Y \mid do(T=0)]$$

- We can also estimate the **average causal effect of treatment on the treated**:

$$\text{ATT} = \mathbb{E}[Y(t=1) - Y(t=0) \mid T=1]$$

- We can also estimate the **average causal effect of treatment on the control:**

$$\text{ATC} = \mathbb{E}[Y(t=1) - Y(t=0) \mid T=0]$$

Estimands for binary treatments

- We generally cannot estimate **unit-level causal effect:** $Y_i(t=1) - Y_i(t=0)$
- We can estimate the average causal effect/**average treatment effect**

$$\mathrm{ATE} = \mathbb{E}[Y(t=1) - Y(t=0)] = \mathbb{E}[Y \mid \mathrm{do}(T=1)] - \mathbb{E}[Y \mid \mathrm{do}(T=0)]$$

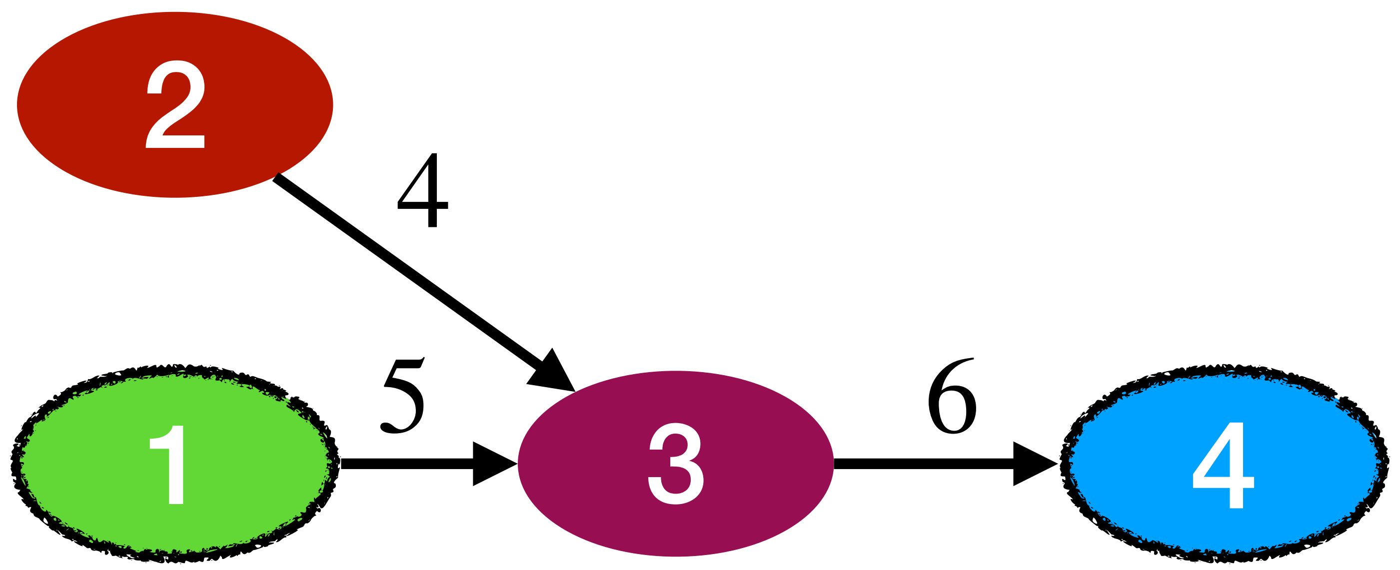
- We can estimate the **average causal effect of treatment on the treated**:

$$\mathrm{ATT} = \mathbb{E}[Y(t=1) - Y(t=0) \mid T=1]$$

- For all, we assume that our covariates $\mathbf{X}$ form a valid adjustment set (e.g. we can check them/filter them with backdoor criterion)

Average causal effect/average treatment effect (ATE)

- $\text{ATE} = \mathbb{E}[Y(t=1) - Y(t=0)] = \mathbb{E}[Y \mid \text{do}(T=1)] - \mathbb{E}[Y \mid \text{do}(T=0)]$



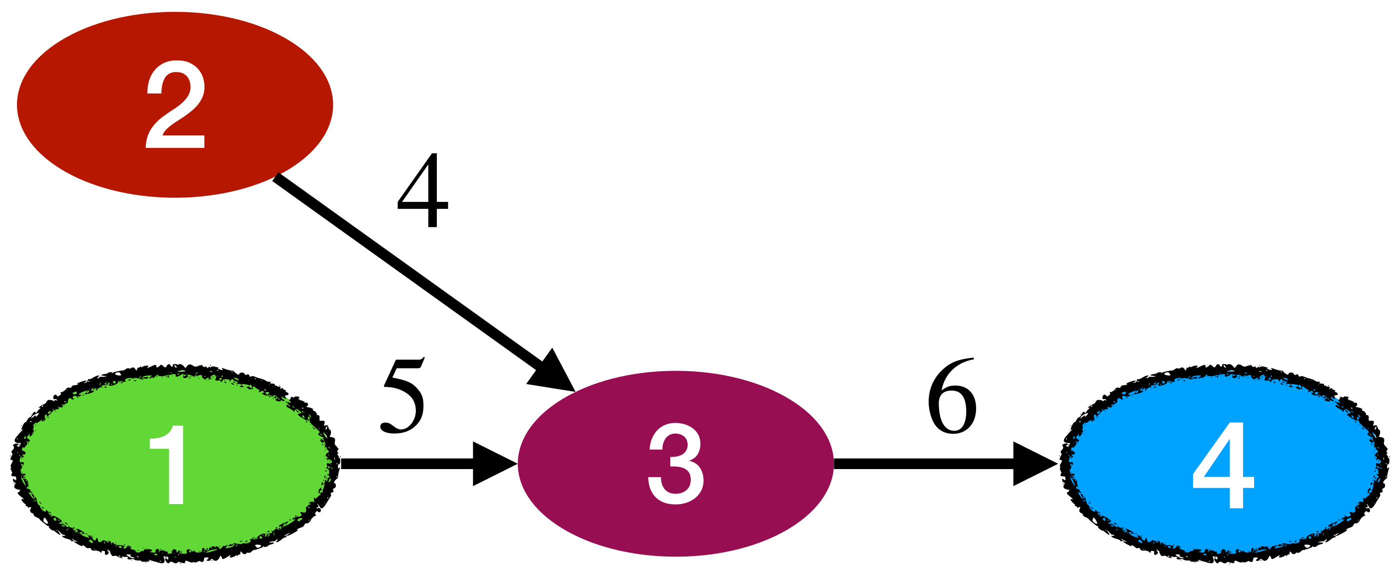

```
x2_1 = randn(n_samples)
x1_1 =  1
x3_1 = 5 * x1_1 + 4 * x2_1 + randn(n_samples)
x4_1 = 6 * x3_1 + randn(n_samples)

x2_0 = randn(n_samples)
x1_0 =  0
x3_0 = 5 * x1_0 + 4 * x2_0 + randn(n_samples)
x4_0 = 6 * x3_0 + randn(n_samples)
diff = np.mean(x4_1) - np.mean(x4_0)
print(diff)
```

```
30.514748479180785
```

Average causal effect/average treatment effect (ATE)

- $\text{ATE} = \mathbb{E}[Y(t=1) - Y(t=0)] = \mathbb{E}[Y \mid \text{do}(T=1)] - \mathbb{E}[Y \mid \text{do}(T=0)]$




```
x2_1 = randn(n_samples)
x1_1 =  1
x3_1 = 5 * x1_1 + 4 * x2_1 + randn(n_samples)
x4_1 = 6 * x3_1 + randn(n_samples)

x2_0 = randn(n_samples)
x1_0 =  0
x3_0 = 5 * x1_0 + 4 * x2_0 + randn(n_samples)
x4_0 = 6 * x3_0 + randn(n_samples)
diff = np.mean(x4_1) - np.mean(x4_0)
print(diff)
```

```
30.514748479180785
```

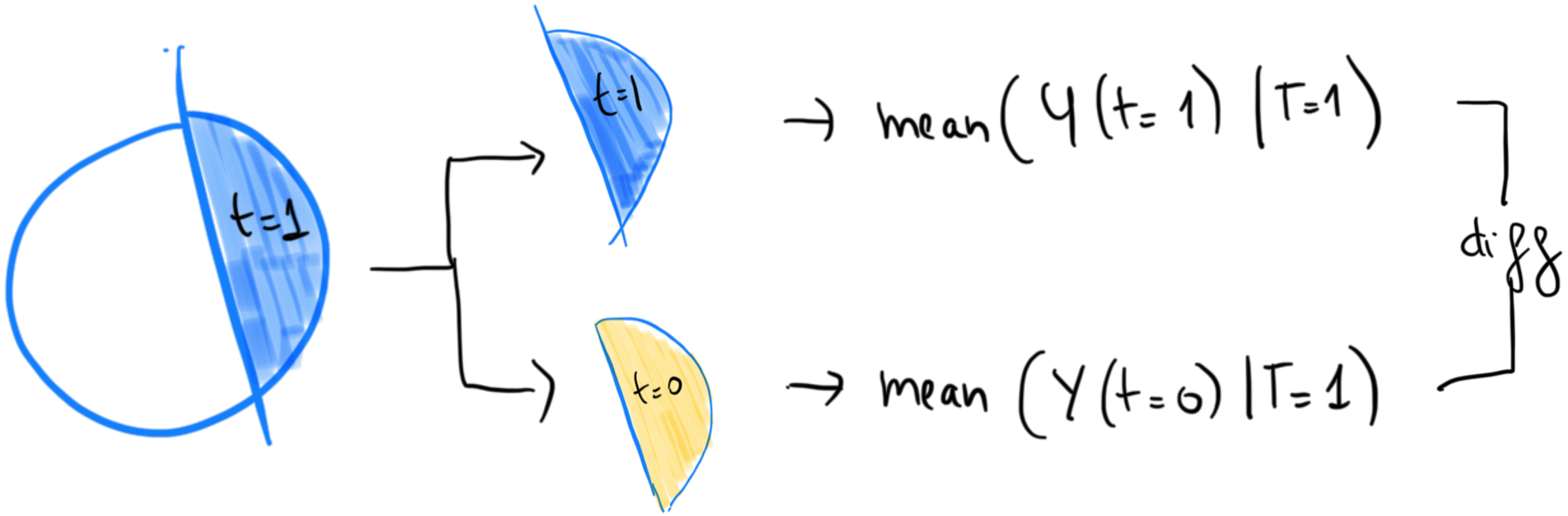
- How well does the treatment work on the patients who choose it?

$\text{ATT} = \mathbb{E}[Y(t=1) - Y(t=0) \mid T=1]$ cannot write in do() notation

Average causal effect of treatment on the treated (ATT)

- How well does the treatment work on the patients who choose it?

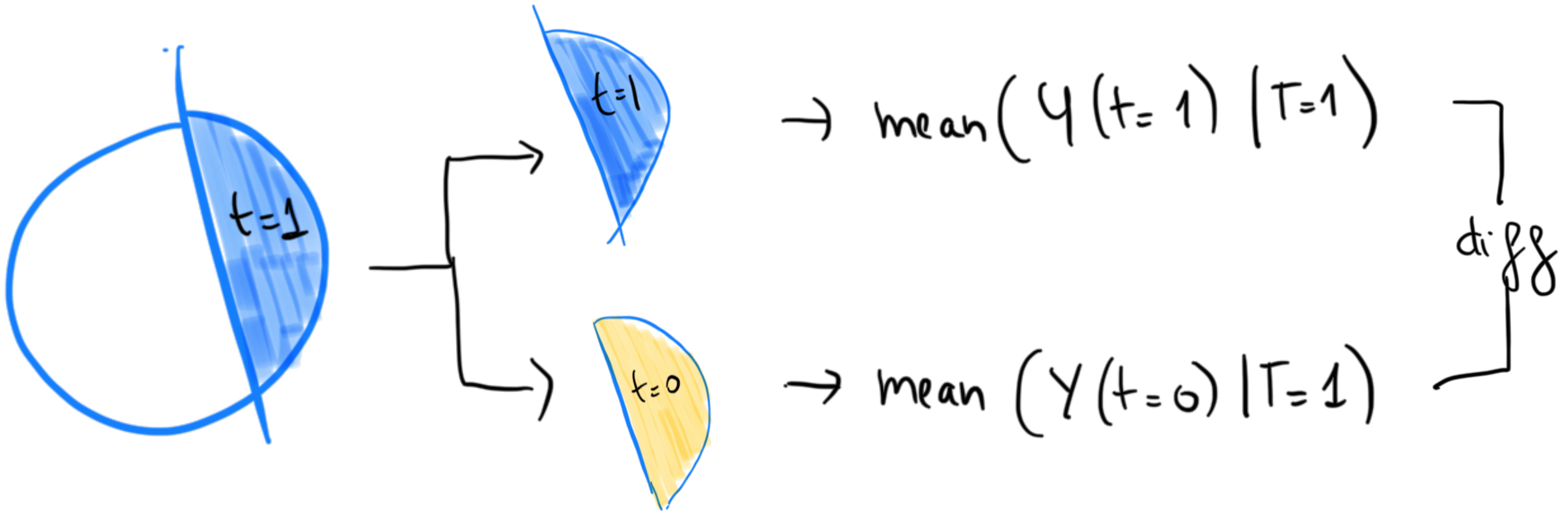
$$\mathrm{ATT} = \mathbb{E}[Y(t=1) - Y(t=0) \mid T=1]$$



Average causal effect of treatment on the treated (ATT)

- How well does the treatment work on the patients who choose it?

$$\mathrm{ATT} = \mathbb{E}[Y(t=1) - Y(t=0) \mid T=1]$$



- **Not the same as ATE:** For example, people who choose a treatment could be more health-conscious, which means they get anyway better outcomes

Estimation method: Matching

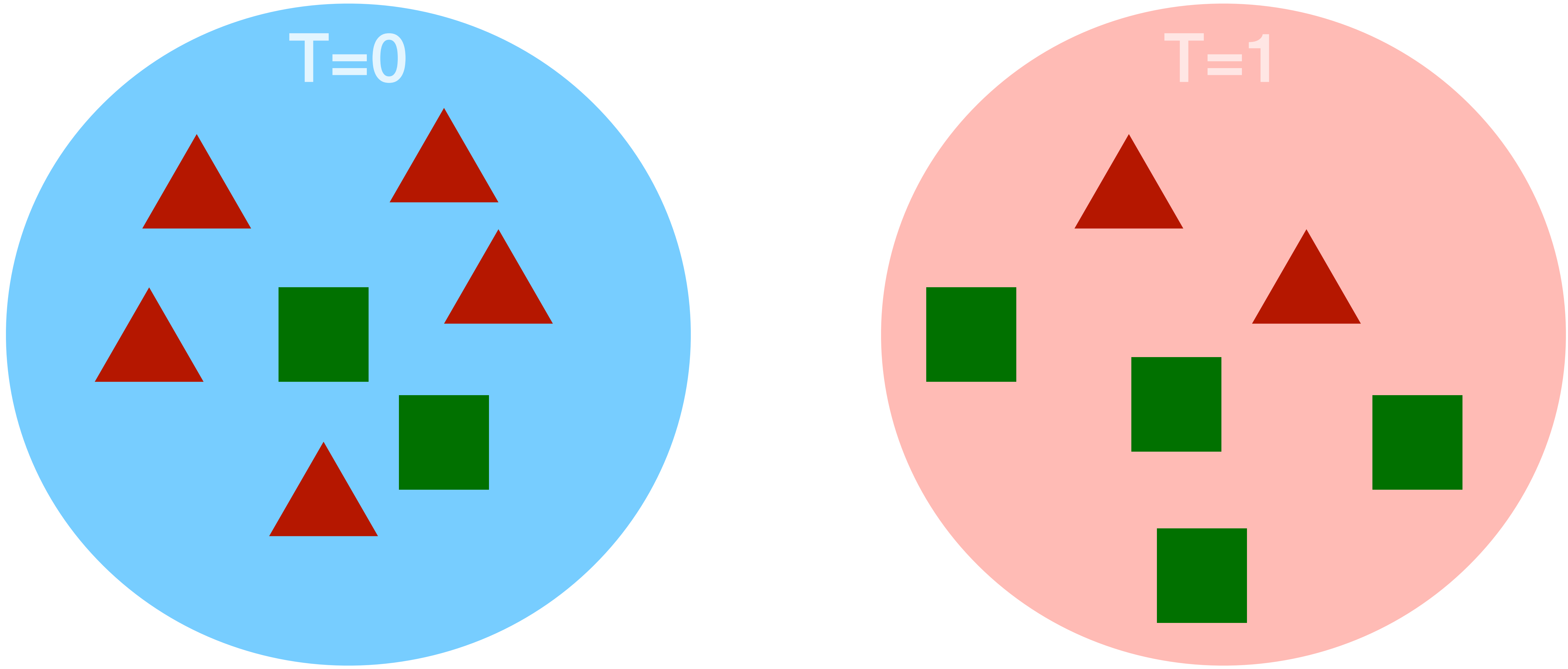
- **Usually for ATT,** sometimes for ATE
- **Intuition:** find the most similar couple of patients in terms of covariates $\mathbf{X}$, such that one is in the treatment and the other in the control group

Estimation method: Matching

- **Usually for ATT,** sometimes for ATE
- **Intuition:** find the most similar couple of patients in terms of covariates $\mathbf{X}$, such that one is in the treatment and the other in the control group
  - If successful, it's like an RCT
  - For example: I want to compare the outcomes of other people of my age

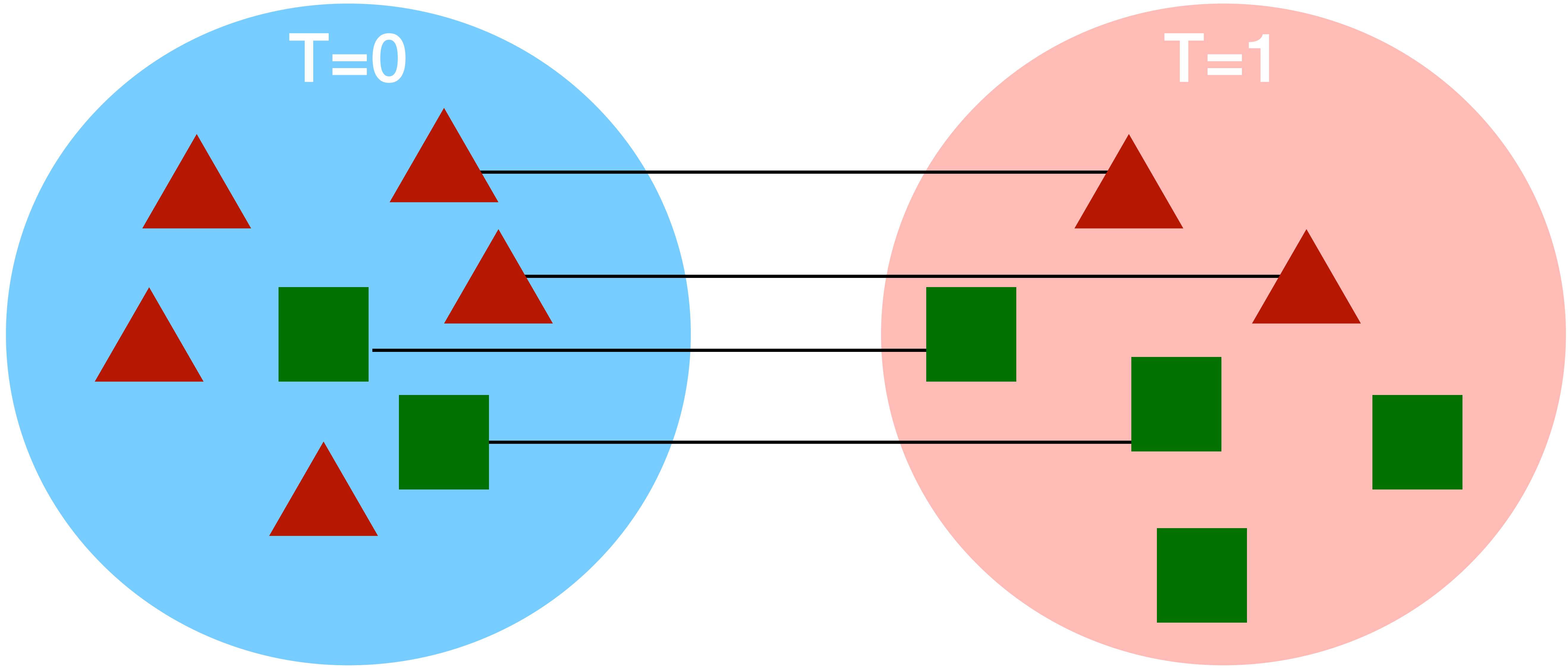
Estimation method: Matching

- **Usually for ATT,** sometimes for ATE
- **Intution:** find the most similar couple of units in terms of covariates $\mathbf{X}$, such that one is in the treatment and the other in the control group



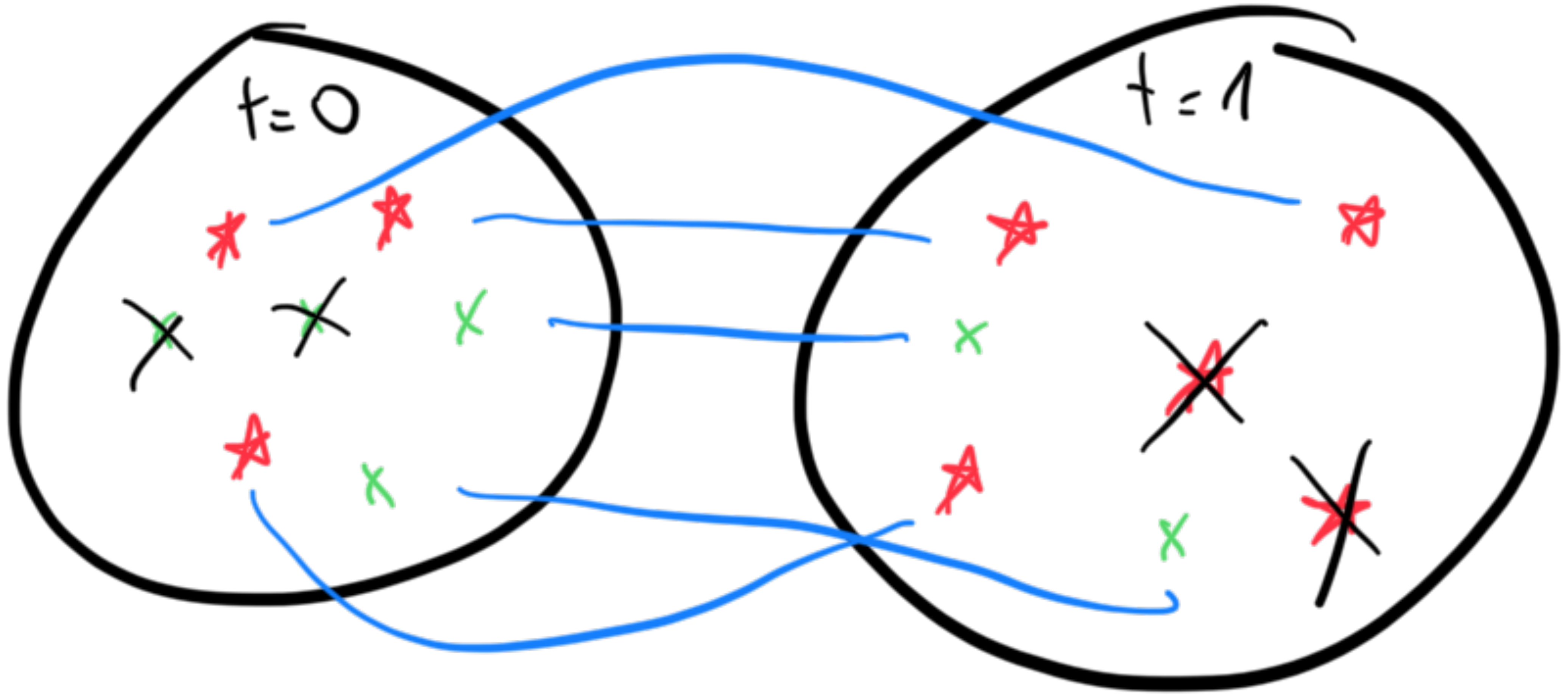
Estimation method: Matching

- **Usually for ATT,** sometimes for ATE
- **Intution:** find the most similar couple of units in terms of covariates $\mathbf{X}$, such that one is in the treatment and the other in the control group



Estimation method: Matching

- **Usually for ATT,** sometimes for ATE
- **Intuition:** find the most similar couple of patients in terms of covariates $\mathbf{X}$, such that one is in the treatment and the other in the control group
  - If successful, it's like an RCT
  - For example: I want to compare the outcomes of other people of my age



- kNN
- Covariate balancing

$T \perp\!\!\!\perp \mathbf{X} \equiv P(\mathbf{X} \mid T=0) = P(\mathbf{X} \mid T=1)$

Exact matching

- **Intuition:** find the most similar couple of patients in terms of covariates $\mathbf{X}$, such that one is in the treatment and the other in the control group
  - For example: I want to compare the outcomes of other people of my age

- **Note:** we can only match units on variables we are adjusting:
  - Units with same values $\mathbf{X} = \mathbf{x}$ in each group are indistinguishable

- **Goal:** discard unmatched units, so we have the same number of units with the same combination of values for $\mathbf{X}$ in treatment and control **(balancing)**

Matching - continuous covariates, greedy/optimal

- If exact matching on the value is not possible, e.g. because we have continuous covariates, we can use any **distance**, e.g Mahalanobis distance

Matching - continuous covariates, greedy/optimal

- If exact matching on the value is not possible, e.g. because we have continuous covariates, we can use any **distance**, e.g Mahalanobis distance

- Many variants exist, in general two types of algorithms:
  - **Greedy matching:** greedily and incrementally match treated with control based on distance
  - **Optimal matching:** optimize for the smallest total distance, can be slow

Matching - continuous covariates, greedy/optimal

- If exact matching on the value is not possible, e.g. because we have continuous covariates, we can use any **distance**, e.g Mahalanobis distance

- Many variants exist, in general two types of algorithms:
  - **Greedy matching:** greedily and incrementally match treated with control based on distance
  - **Optimal matching:** optimize for the smallest total distance, can be slow

- Need to check **covariate balancing** after matching (e.g. std mean difference)

$$T \perp\!\!\!\perp \mathbf{X} \equiv P(\mathbf{X} \mid T = 0) = P(\mathbf{X} \mid T = 1)$$

Estimation method: Propensity score matching (PSM)

- **Assumptions**: binary treatment T, $\mathbf{X}$ is valid adjustment set
- **Propensity score:** the probability of getting assigned the treatment

$$e(x) \quad \pi(x) := P(T = 1 \mid \mathbf{X} = x)$$

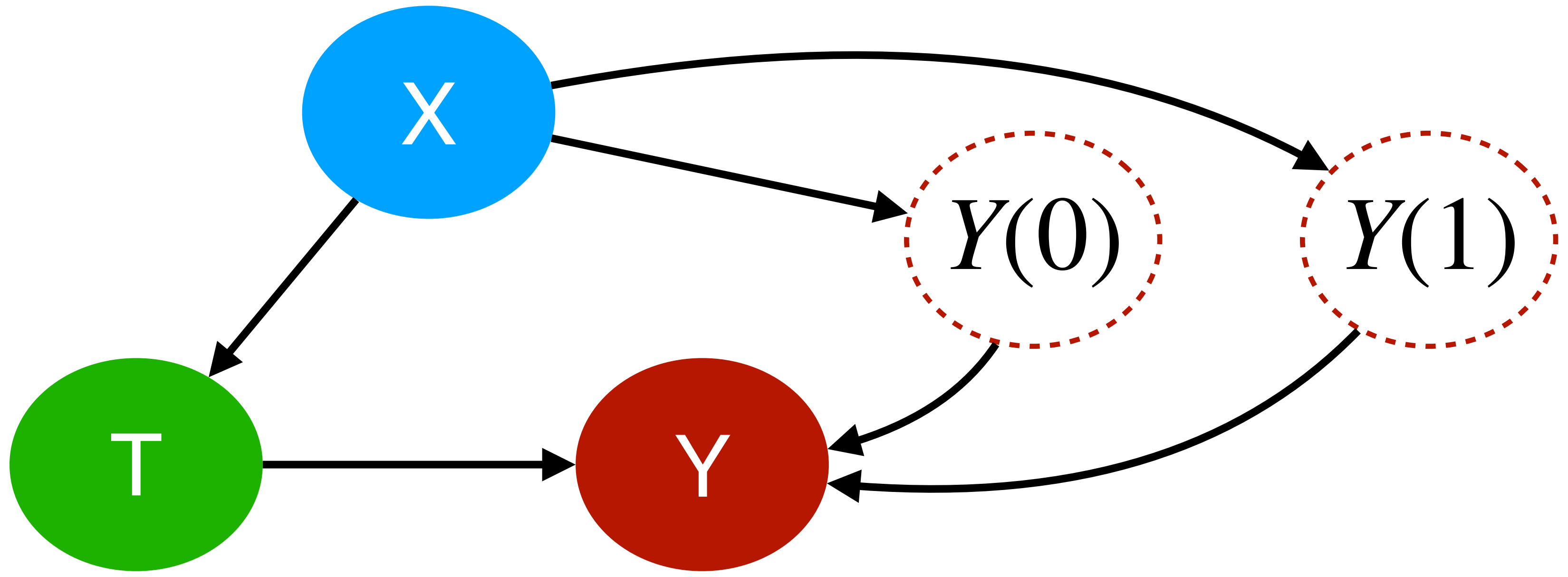
Estimation method: Propensity score matching (PSM)

- **Assumptions**: binary treatment T, $\mathbf{X}$ is valid adjustment set
- **Propensity score:** the probability of getting assigned the treatment

$$e(x) \quad \pi(x) := P(T = 1 \mid \mathbf{X} = x)$$

Conditional ignorability/No unmeasured confounding

- We can show that $T \perp\!\!\!\perp \mathbf{X} \mid \pi(\mathbf{X})$ and that if $Y(0), Y(1) \perp\!\!\!\perp T \mid \mathbf{X}$ then



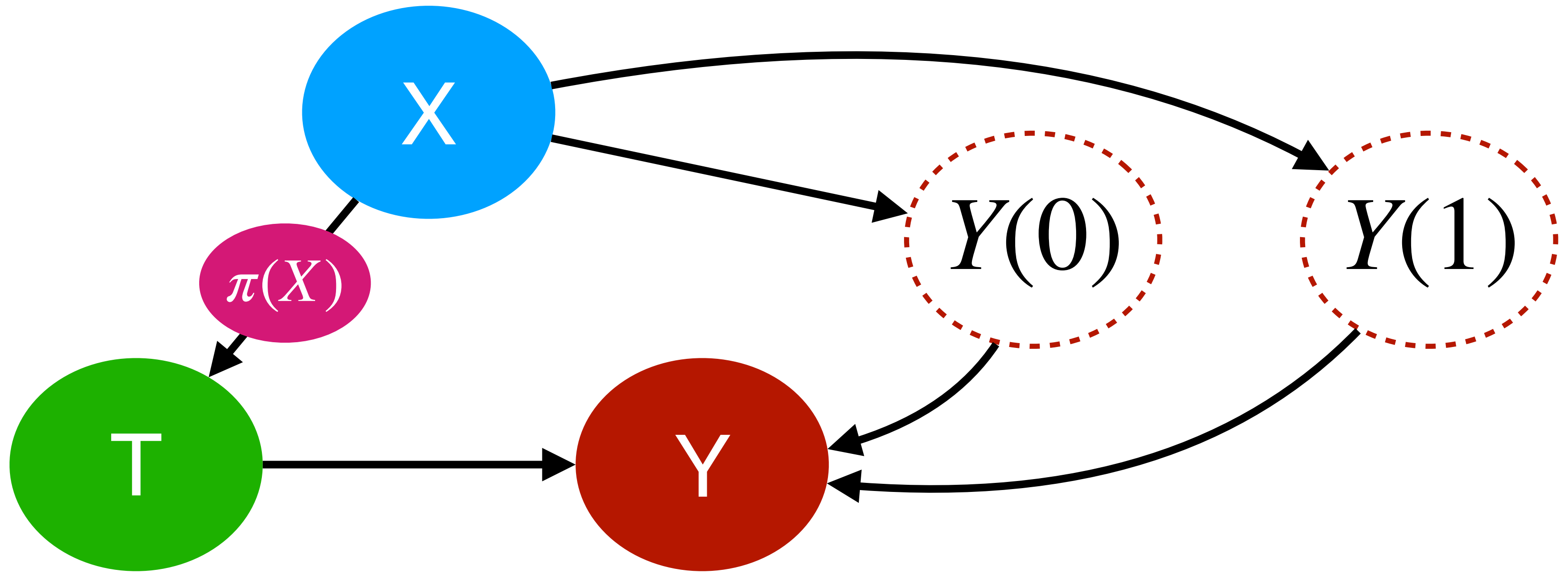
Estimation method: Propensity score matching (PSM)

- **Assumptions**: binary treatment T, $\mathbf{X}$ is valid adjustment set
- **Propensity score:** the probability of getting assigned the treatment

$$e(x) \quad \pi(x) := P(T = 1 \,|\, \mathbf{X} = x)$$

- We can show that $T \perp\!\!\!\perp \mathbf{X} \,|\, \pi(\mathbf{X})$ and that if $Y(0), Y(1) \perp\!\!\!\perp T \,|\, \mathbf{X}$ then

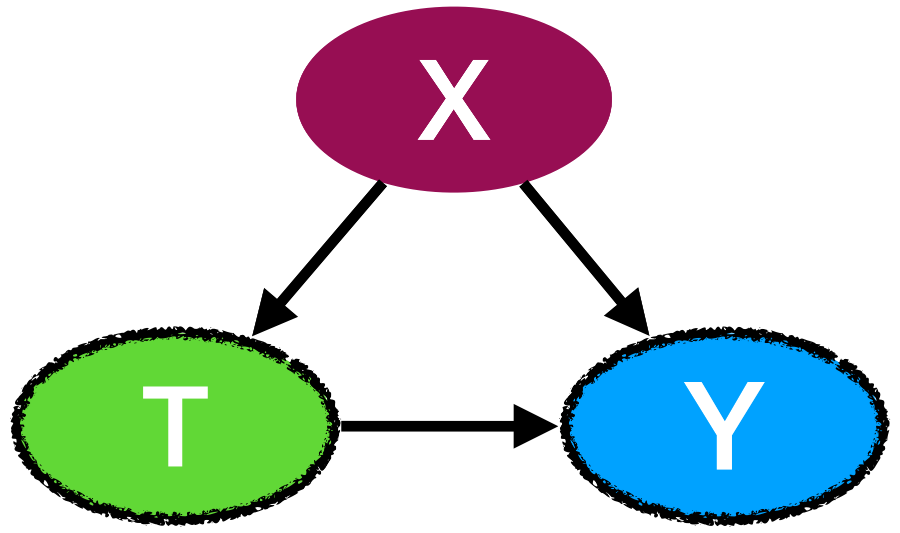
$$Y(0), Y(1) \perp\!\!\!\perp T \,|\, \pi(\mathbf{X})$$



e.g. with **logistic regression**

- We can estimate π from data and use it to match
  - If $\mathbf{X}$ has a lot of covariates, it is easier to match since it's a single number

Matching and IPW Jupyter notebook



$P(\mathbf{X} = 1) = 0.3$

$P(T = 1 \mid \mathbf{X} = 1) = 0.1$

$P(T = 1 \mid \mathbf{X} = 0) = 0.9$

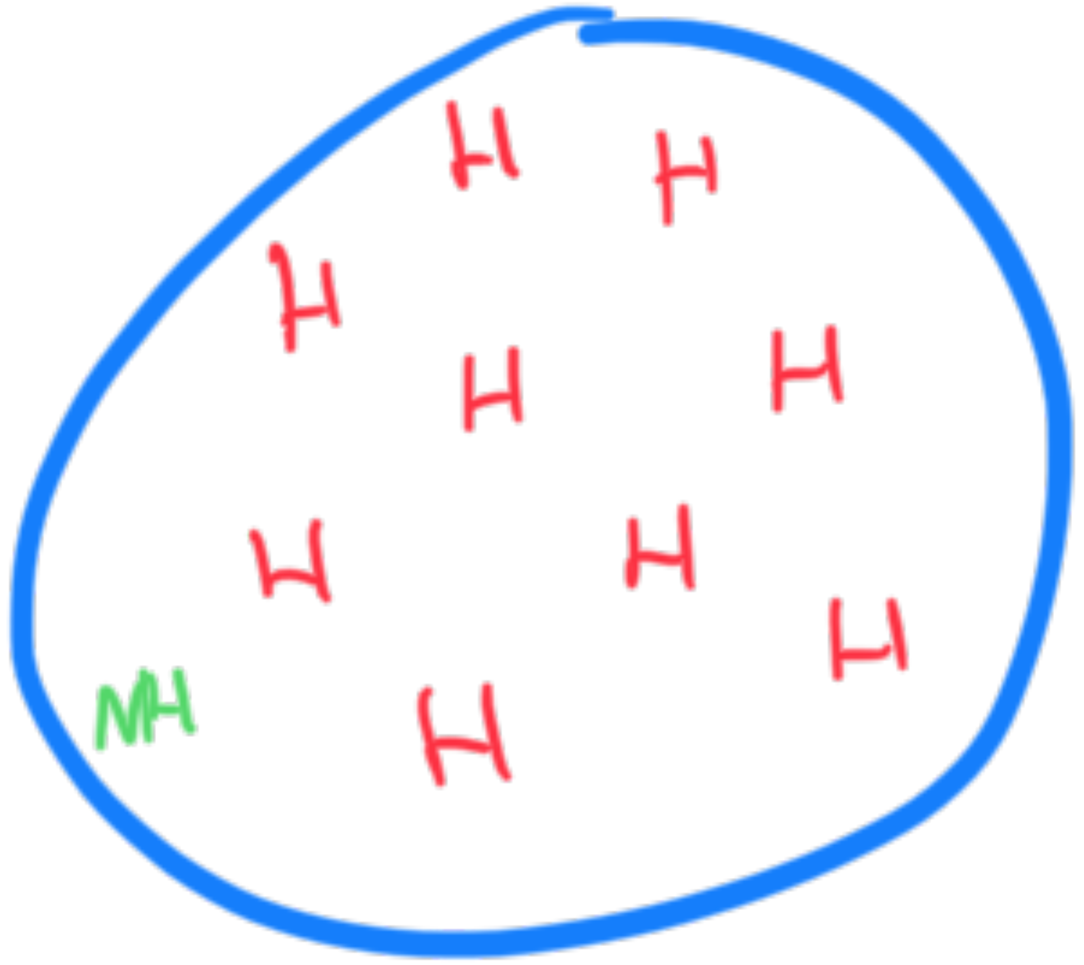
$P(Y = 1 \mid T = 1, \mathbf{X} = 1) = 0.75$

$P(Y = 1 \mid T = 0, \mathbf{X} = 1) = 0.5$

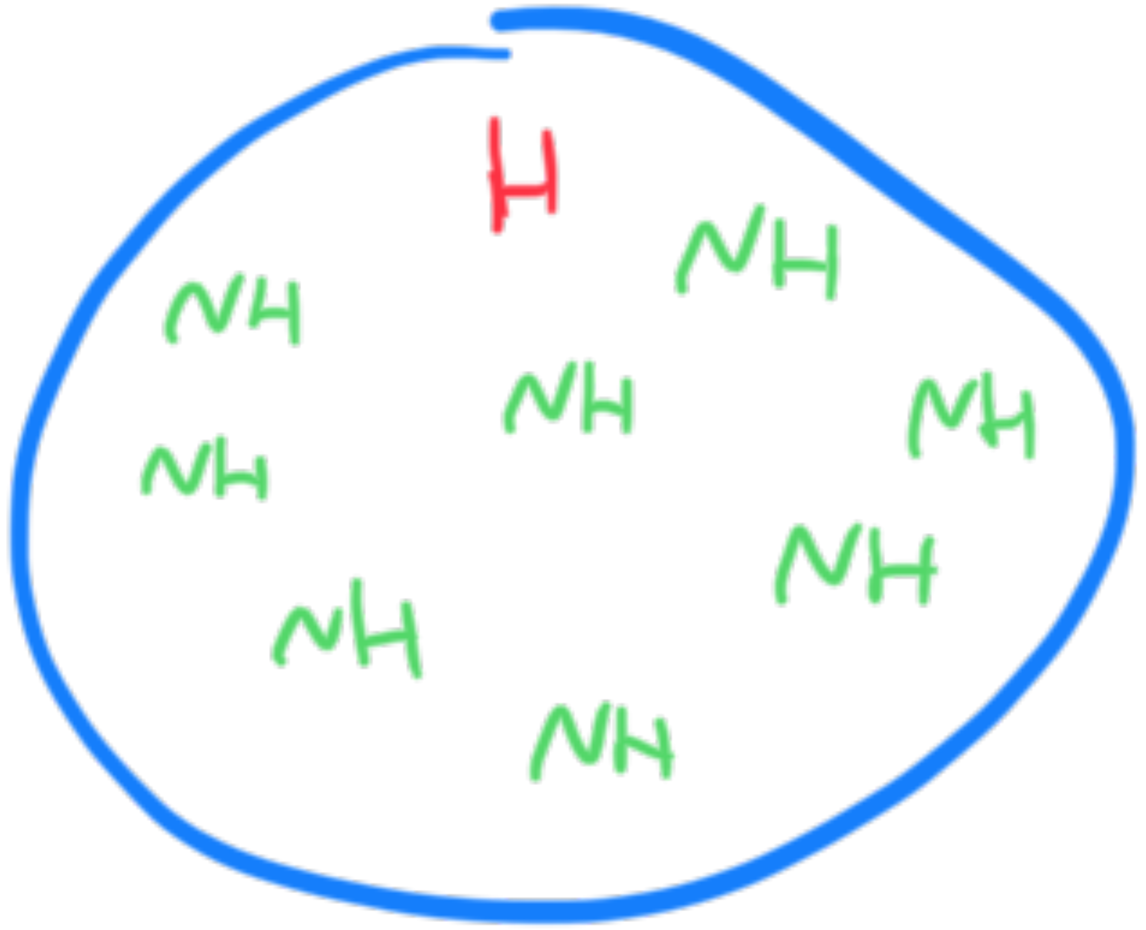
$P(Y = 1 \mid T = 0, \mathbf{X} = 0) = 0.6$

$P(Y = 1 \mid T = 1, \mathbf{X} = 0) = 0.9$

T=0



T=1



Matching and IPW Jupyter notebook

```
treatment_group_x_0 = treatment_group[treatment_group["x"]==0]
treatment_group_x_1 = treatment_group[treatment_group["x"]==1]

control_group_x_0 = control_group[control_group["x"]==0]
control_group_x_1 = control_group[control_group["x"]==1]

print("Number of people with X=0 in treatment: ", len(treatment_group_x_0)," and in control: ", len(control_group_x_0))
print("Number of people with X=1 in treatment: ", len(treatment_group_x_1)," and in control: ", len(control_group_x_1))
```

```
Number of people with X=0 in treatment:  3159  and in control:  338
Number of people with X=1 in treatment:  157  and in control:  1346
```

$$P(\mathbf{X} = 1) = 0.3 \qquad P(T = 1 \,|\, \mathbf{X} = 1) = 0.1 \qquad P(T = 1 \,|\, \mathbf{X} = 0) = 0.9$$

```
min_number_x1 = min(len(treatment_group_x_1), len(control_group_x_1))
balanced_treatment_x_1 = treatment_group_x_1[0:min_number_x1]
balanced_control_x_1 = control_group_x_1[0:min_number_x1]
print("After balancing: number of people with X=1 in treatment: ", len(balanced_treatment_x_1)," and in control: ", len(balanced_control_x_1))
```

```
After balancing: number of people with X=0 in treatment:  338  and in control:  338
After balancing: number of people with X=1 in treatment:  157  and in control:  157
```

Next class: Inverse probability weighting (IPW)

- **Idea:** rather than match, reweight (downweight or upweight) observations
- **Inverse probability (of treatment) weighting:** weight by inverse of probability of treatment **received**:
    - For treated $T = 1$: weight by the inverse of $\pi = P(T = 1 \mid \mathbf{X})$
    - For untreated $T = 0$: weight by the inverse of $1 - \pi = P(T = 0 \mid \mathbf{X})$

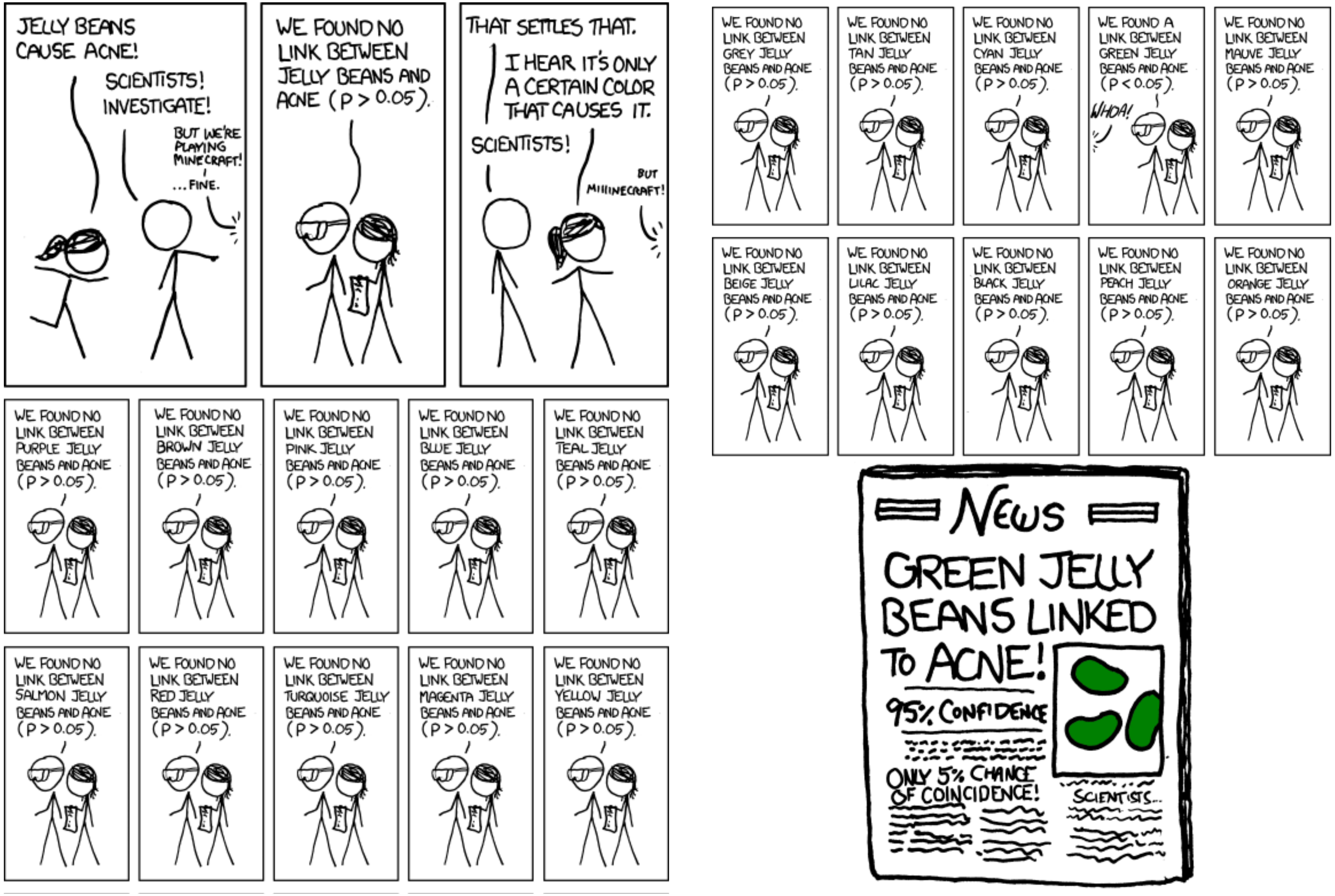
Next class: Inverse probability weighting (IPW)

- **Inverse probability (of treatment) weighting:** weight by inverse of probability of treatment **received**:
  - For treated $T = 1$: weight by the inverse of $\pi = P(T = 1 \mid \mathbf{X})$
  - For untreated $T = 0$: weight by the inverse of $1 - \pi = P(T = 0 \mid \mathbf{X})$

$$\hat{\mathbb{E}}(Y(t=1)) = \frac{1}{n}\sum_{i=1}^{n} Y_i \cdot 1\{T=1\} \cdot \frac{1}{P(T=1 \mid X_i)}$$

$$\hat{\mathbb{E}}(Y(t=0)) = \frac{1}{n}\sum_{i=1}^{n} Y_i \cdot 1\{T=0\} \cdot \frac{1}{P(T=0 \mid X_i)}$$

Questions?



https://xkcd.com/882/