DS 102 Homework 4

If you are handwriting your solution please make sure your answers are legible, as you may lose points otherwise.
Data science is a collaborative activity. While you may talk with others about the homework, please write up your solutions individually. If you do discuss the homework with others, please include their names on your submission.
Due on Gradescope by 9:29am, Tuesday 7th April, 2020

1. (15 points) **Backdoor criteria**
   A research team wants to estimate the effectiveness of a new veterinary drug for sick seals. They ask aquariums across the country to volunteer their sick seals for the experiment. Since the team offers monetary compensation for volunteering, zoos with less income decide to volunteer their sick seals, whereas zoos with more income are less compelled to volunteer their seals.

   It turns out that zoos with less income feed their seals less nutritious diets (regardless of whether they are sick or healthy), due to budgetary constraints. Less nutritious diets prevent seals from recovering as effectively.

   (a) (5 points) **Draw** a causal graph between variables X, Y, I and N which denote receiving the drug, recovering, the income level of the zoo, and how nutritious a seal's diet is, respectively. Justify each edge in your graph.

   (b) (5 points) The *backdoor criterion* is a criterion for determining which variables we can adjust for, to block all the confounding pathways between X and Y. Formally, in a causal graph, we define a path between two nodes X and Y as a sequence of nodes beginning with X and ending with Y, in which each node is connected to the next by an edge (which can have either direction). Given an ordered pair of variables (X, Y), a set of variables Z satisfies the backdoor criterion relative to (X, Y) if no node in Z is a descendant of X, and Z blocks every path between X and Y that contains an arrow into X.

   Using the causal graph in the previous part, **determine** a minimal set of variable(s) that satisfies the backdoor criterion relative to (X, Y). (The answer may not be unique.)

   (c) (5 points) Given your answer to the previous part, **what is the adjustment formula** for the effect of the drug on recovery?

   Hint: see Lecture 14 Section 14.2.

2. (20 points) **Inverse Propensity Score Weighting** In this problem, we'll develop and implement an estimator for the treatment effect $\mathbb{E}[Y \mid \mathbf{do}(X := 1)] - \mathbb{E}[Y \mid \mathbf{do}(X := 0)]$, where Y is the outcome variable and X is the treatment variable.

   To do this, we'll use inverse propensity score weighting (see Lecture 14). Let

$$e(z) = \mathbb{E}[X \mid Z = z] = \mathbb{P}(X = 1 \mid Z = z)$$

denote the probability that the treatment is administered, given the value of the confounders. When Z denotes all confounders (*i.e.* there are no hidden confounders), we have

$$\mathbb{E}[Y \mid \mathbf{do}(X := 1)] = \mathbb{E}\left[\frac{YX}{e(Z)}\right]$$

which motivates a practical estimator for $\mathbb{E}[Y \mid \mathbf{do}(X := 1)]$, described as follows.

Suppose we have a dataset with n samples, where $x_i \in \{0, 1\}$, y_i, and z_i are the values of the treatment, outcome, and confounders for the i-th sample, respectively (note that the treatment is binary-valued, since it was either administered or not).

In practice, we don't know $e(z)$, but we can estimate it by fitting a logistic regression model $\hat{e}(z) \approx e(z)$ that predicts x_i from z_i. We can then use the following estimator for $\mathbb{E}[Y \mid \mathbf{do}(X := 1)]$:

$$\frac{1}{n}\sum_{i=1}^{n} \frac{x_i y_i}{\hat{e}(z_i)}$$

which we call the *inverse propensity score weighted estimator* (IPSWE).

(a) (10 points) Show that $\mathbb{E}[Y \mid \mathbf{do}(X := 0)] = \mathbb{E}[\frac{Y(1-X)}{1-e(Z)}]$. (Hint: follow the derivation of $\mathbb{E}[Y \mid \mathbf{do}(X := 1)] = \mathbb{E}[\frac{YX}{e(Z)}]$ in Lecture 14 Section 14.3, modifying as necessary to model $\mathbf{do}(X := 0)$ instead of $\mathbf{do}(X := 1)$.)

(b) (5 points) Write an IPSWE for $\mathbb{E}[Y \mid \mathbf{do}(X := 0)]$ that uses an estimated $\hat{e}(z) \approx e(z)$, analogous to the IPSWE of $\mathbb{E}[Y \mid \mathbf{do}(X := 1)]$.

(c) (5 points) Write an estimator for the treatment effect. (Hint: Combine the IPSWEs of $\mathbb{E}[Y \mid \mathbf{do}(X := x)]$, $x \in \{0, 1\}$.)

3. (25 points) **Causal Inference on an IHDP Dataset**

In this problem, you'll apply the treatment effect estimator derived in the previous problem to a dataset from the Infant Health and Development Program (IHDP). The IHDP was an experiment that treated low-birth-weight, premature infants with intensive high-quality childcare from a trained provider. The goal will be to estimate the causal effect of this treatment on the outcome, the children's cognitive test scores. This dataset *does not* represent a randomized trial in which treatments were randomly assigned, so there may be confounders between the treatment and outcome.

Download the dataset the course website (`ihdp.csv`). You can do this problem in a Jupyter Notebook, then save and upload it as a PDF, just as you do with lab assignments. Please include all code, plots, and code used to generate plots with your submission.

(a) (5 points) The CSV file `ihdp.csv` has 27 columns:

- Column 1 is the treatment $x_i \in \{0, 1\}$, which indicates whether or not the treatment was given to the infant.
- Column 2 is the outcome $y_i \in \mathbb{R}$, the child's cognitive test score.
- Columns 3-27 contain 25 features of the mother and child (*e.g.* the child's birth weight, whether or not the mother smoked during pregnancy, her age and race). Since this dataset was not collected by a randomized trial, these features could all confound x_i and y_i, and are denoted by $z_i \in \mathbb{R}^{25}$.

In this part, you'll estimate $\hat{e}(z)$ by fitting a logistic regression model that predicts x_i from z_i. For any z_i, $\hat{e}(z_i)$ is then the predicted probability that $x_i = 1$ made by this logistic regression model on z_i. Specifically:

1. Read the data in `ihdp.csv` (*e.g.* using the `csv` package in Python) into three arrays: $X \in \{0, 1\}^n$ containing the treatments, $Y \in \mathbb{R}^n$ containing the outcomes, and $Z \in \mathbb{R}^{n \times 25}$ containing the features.
2. To fit a logistic regression model, use the `scikit-learn` package in Python, which is imported as `sklearn`. Start with the following two lines:
```
from sklearn.linear_model import LogisticRegression as
LR
lr = LR(penalty='none', max_iter=200, random_state=0)
```
3. Use the `lr.fit()` method to fit the logistic regression model $\hat{e}(z)$ (see the documentation here.)

(b) (10 points) Write a function `estimate_treatment_effect` that computes the estimator of the treatment effect you derived in the previous problem. It should take in the following arguments:

- a fitted linear regression model (the `LogisticRegression` object `lr` from the previous part)
- X
- Y
- Z

and output a single value, which is the estimate of the treatment effect. (Hint: See the `LogisticRegression` object's `predict_proba` method.)

(c) (5 points) Use the function `estimate_treatment_effect` from the previous part to estimate the treatment effect for the IHDP dataset. Report this estimate. According to the estimate, did the treatment have a beneficial causal effect on the outcome (*i.e.* cause cognitive test scores to increase)?

(d) (5 points) The difference between the empirical conditional expectations,

$$\frac{1}{n_1}\sum_{i=1}^{n} y_i \mathbb{1}[x_i = 1] - \frac{1}{n_0}\sum_{i=1}^{n} y_i \mathbb{1}[x_i = 0] \approx \mathbb{E}[Y \mid X = 1] - \mathbb{E}[Y \mid X = 0],$$

where $n_1 = \sum_{i=1}^n \mathbb{1}[x_i = 1]$ and $n_0 = \sum_{i=1}^n \mathbb{1}[x_i = 0]$ is a "naive" estimator of the treatment effect. Report this estimate on the IHDP dataset. Why is it different from the estimate you computed in the previous part? Are there any circumstances under which these two estimators should produce the same estimates?

4. (40 points) **Regret of the Explore-then-Commit Algorithm** In this problem, we will analyze the regret of the Explore-then-Commit algorithm for a multi-armed-bandit problem.

   Suppose we have a stochastic multi-armed bandit problem where there are K arms. Let P_i denote the reward distribution of arm i, which has mean μ_i. At round t, each arm independently generates a reward from its reward distribution. Let $Y_{i,t} \sim P_i, i = 1, \dots, K$ denote the reward of arm i in round t. Assume the rewards are bounded between a and b for some $a < b$, *i.e.* $\mathbb{P}(Y_{i,t} \in [a, b]) = 1$ for $Y_{i,t} \sim P_i$, $i = 1, \dots, K$ and for all t.

   At round t, the player pulls an arm $A_t \in \{1, \dots, K\}$. The reward actually received by the player on round t is then $X_t = Y_{A_t,t}$.

   See Algorithm 1 for the Explore-then-Commit (EC) algorithm. EC iterates through the K arms, and pulls each arm c times for a total of cK pulls. After cK pulls, it commits to the arm with the highest sample mean $\hat{\mu}_i$:

$$\hat{\mu}_i = \frac{1}{c} \sum_{s=1}^{c} Y_{i,K(s-1)+i}$$

   where $Y_{i,K(s-1)+i}$ is equivalent to the reward received when the i-th arm is pulled the s-th time. The algorithm then pulls that arm (with the highest sample mean) every time afterwards.

Algorithm 1 Explore-then-Commit Algorithm

input: Number of exploratory pulls c per arm

For $t = 1, 2, \dots$:

$$A_t = \begin{cases} (t \bmod k) + 1 & : \quad t \leq cK \\ \arg\max_{i \in \{1,\dots,K\}} \hat{\mu}_i & : \quad t > cK \end{cases}$$

   We define the mean of the optimal arm as

$$\mu^* = \max_{i \in \{1,\dots,K\}} \mu_i$$

   and the *sub-optimality gap* of a sub-optimal arm i as

$$\Delta_i = \mu^* - \mu_i.$$

In this problem, we will analyze the *pseudo-regret* of this algorithm. The pseudo-regret of an algorithm is given by:

$$R(n) = n\mu^* - \mathbb{E}\left[\sum_{t=1}^{n} X_t\right].$$

(a) (10 points) Define the random variable $T_i(t)$ as the number of times arm i has been pulled, up to and including time n:

$$T_i(n) = \sum_{t=1}^{n} \mathbb{I}\{A_t = i\}.$$

Show that we can decompose the regret as:

$$R(n) = \sum_{i=1}^{K} \Delta_i \mathbb{E}[T_i(n)].$$

Hint: Start with the following:

$$\begin{aligned} n\mu^* - \mathbb{E}\left[\sum_{t=1}^{n} X_t\right] &= \mathbb{E}\left[\sum_{t=1}^{n} (\mu^* - X_t)\right] \\ &= \mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} \mathbb{I}\{A_t = i\}(\mu^* - Y_{i,t}).\right] \end{aligned}$$

(Make sure you understand these lines.) Note also that for all t, A_t is independent of $Y_{i,t}, i = 1, \ldots, K$.

(b) (5 points) Show that if $n > cK$, then

$$\mathbb{E}[T_i(n)] = c + (n - Kc)\mathbb{P}\left(\hat{\mu}_i > \max_{j=1,\ldots,K, j\neq i} \hat{\mu}_j\right)$$

Hint: If $n > cK$, every arm is pulled deterministically c times. Afterward, an arm is only pulled if it is the one with the maximum sample mean $\hat{\mu}_i$.

(c) (5 points) Suppose, without loss of generality, that the optimal arm (the arm with the highest mean μ^*) is the first arm, *i.e.* $\mu^* = \mu_1$. Show that for any sub-optimal arm i:

$$\mathbb{P}\left(\hat{\mu}_i > \max_{j=1,\ldots,K; j\neq i} \hat{\mu}_j\right) \leq \mathbb{P}\left(\hat{\mu}_i > \hat{\mu}_1\right)$$

Hint: Think about the two events that we are looking at the probabilities of. One of these events is a subset of the other.

(d) (10 points) Putting together our results from the last few parts, we have shown that:

$$\mathbb{E}[T_i(n)] \leq c + (n - Kc)\mathbb{P}\left(\hat{\mu}_i > \hat{\mu}_1\right).$$

Using the Hoeffding bound, show that

$$\mathbb{P}\left(\hat{\mu}_i > \hat{\mu}_1\right) = \mathbb{P}\left(\frac{1}{c}\sum_{s=1}^{c} Y_{i,K(s-1)+i} > \frac{1}{c}\sum_{s=1}^{c} Y_{1,K(s-1)+i}\right) \leq \exp\left(-\frac{c\Delta_i^2}{2(b-a)^2}\right)$$

(Hint: The Hoeffding bound applies to random variables $Z_1, \ldots, Z_m$ where each random variable Z_j is bounded between u_j and l_j for $j = 1, \ldots, m$. The bound then states that

$$\mathbb{P}\left(\sum_{j=1}^{m} Z_j - \mathbb{E}\left[\sum_{j=1}^{m} Z_j\right] > t\right) \leq \exp\left(-\frac{2t^2}{\sum_{j=1}^{m}(u_j - l_j)^2}\right).$$

Recall that $Y_{i,K(s-1)+i}$ is the reward when the i-th arm is pulled the s-th time, and $Y_{i,K(s-1)+i} - Y_{1,K(s-1)+i}$ is bounded between $b-a$ and $a-b$ with mean $\mu_i - \mu_1$.)

(e) (10 points) Putting our results together, we have that:

$$\mathbb{E}[T_i(n)] \leq c + (n - Kc)\exp\left(-\frac{c\Delta_i^2}{2(b-a)^2}\right).$$

Suppose you knew the minimum sub-optimality gap,

$$\Delta = \min_{i>1} \Delta_i.$$

Then for each sub-optimal arm $i = 2, \ldots, K$, we further have

$$\mathbb{E}[T_i(n)] \leq c + (n - Kc)\ \exp\left(-\frac{c\Delta^2}{2(b-a)^2}\right) \leq c + n\ \exp\left(-\frac{c\Delta^2}{2(b-a)^2}\right),$$

where we upper-bounded $n - Kc$ by n.

Solve for a value of c which guarantees that:

$$\exp\left(-\frac{c\Delta^2}{2(b-a)^2}\right) \leq \frac{1}{n}.$$

For this number of exploratory pulls c, what is the upper bound on the pseudo-regret? Your answer should be in terms of $n, a, b, \Delta, \Delta_i$. Does this bound grow linearly in n, or does it do better (is it sublinear)?

Hints: Use the pseudo-regret decomposition you derived in Part (a) and plug in the upper bound on $\mathbb{E}[T_i(n)]$ shown above. Also note that $\Delta_1 = \mu^* - \mu_1 = \mu_1 - \mu_1 = 0$.