

Answers Coding Exercise Module 1

This notebook contains the answers for the coding exercise in Module 1 of the 2025 course “Causal Inference with Linear Regression: A Modern approach” by CausAI.

Imports

```
import pandas as pd
import numpy as np
```

Question 1.1: Load Data

```
data = pd.read_csv("internship_salary_data.csv")
```

data

	academic_performance	internship	starting_salary
0	Medium	1	49213.56
1	High	1	67619.44
2	Medium	0	39499.34
3	Medium	1	34011.98
4	Low	0	37126.74
...
49995	Medium	1	39844.48
49996	Low	0	37547.89
49997	Medium	1	40521.04
49998	Medium	0	38052.10
49999	High	1	60851.96

Question 1.2: Compute $E[\text{starting_salary} | \text{internship} = 1] - E[\text{starting_salary} | \text{internship} = 0]$

```
# Group by internship and calculate mean starting_salary for each group
grouped_on_treatment = (
    data[["internship", "starting_salary"]].groupby(["internship"]).mean()
)

# Extract mean starting_salary for treated and untreated groups
salary_mean_treated = grouped_on_treatment.loc[
    1, "starting_salary"
] # approximation for  $E[\text{starting\_salary} | \text{internship} = 1]$ 
```

```

salary_mean_untreated = grouped_on_treatment.loc[
    0, "starting_salary"
] # approximation for E[starting_salary | internship = 0]

# Calculate the difference
naive_ATE = salary_mean_treated - salary_mean_untreated

print(f"Mean starting salary (Treated): {salary_mean_treated:.4f}")
print(f"Mean starting salary (Untreated): {salary_mean_untreated:.4f}")
print(f"Difference in starting salary (naive ATE estimate): {naive_ATE:.4f}")

```

```

Mean starting salary (Treated): 46589.0023
Mean starting salary (Untreated): 37384.3005
Difference in starting salary (naive ATE estimate): 9204.7018

```

Question 1.3: If we would interpret this as the ATE, we would do this as follows: completing an intership increases someones annual starting salary by 9205 U.S. dollars, on average.

This relies on the implicit assumption that we have ignorability: whether an intership is done or not looks as good as randomly assigned among the individuals.

Question 2.1: The Causal Graph looks like

$$internship \leftarrow academic_performance \rightarrow starting_salary$$

$$internship \rightarrow starting_salary$$

Question 2.2:

Given a pair of variables (T, Y) in a Causal Graph, a set of variables Z satisfies the backdoor criterion relative to (T, Y) if Z blocks every path between T and Y that contains an arrow into T and no node in Z is a descendant of T .

So in our graph, the path $internship \leftarrow academic_performance \rightarrow starting_salary$ must be blocked.

And recall that a path is blocked by a set of variables Z if

1. There is a pattern $\dots \leftarrow A \rightarrow \dots$ or $\dots \rightarrow A \rightarrow \dots$ where A is in Z
2. The path contains a collider that isn't in Z and none of its descendants are in Z

In this case, we have a simple fork pattern $internship \leftarrow academic_performance \rightarrow starting_salary$, which is blocked by conditioning on $academic_performance$

Furtermore, $internship$ doesn't have any descendants in the graph.

Therefore $\{academic_performance\}$ satisfies the backdoor criterion, and we have conditional ignorability given $academic_performance$.

Question 3.1:

```
academic_performance_proportions = (  
    data["academic_performance"].value_counts(normalize=True).to_dict()  
)  
academic_performance_proportions
```

```
{'Medium': 0.5003, 'Low': 0.30118, 'High': 0.19852}
```

Question 3.2:

```
# Calculate average starting salary stratified by academic performance and internship  
stratified_avg_salary = (  
    data.groupby(["academic_performance", "internship"])["starting_salary"]  
    .mean()  
    .reset_index()  
)  
  
# Step 3: Compute the adjusted ATE as a weighted average  
ate_adjusted = 0  
for z, p_z in academic_performance_proportions.items():  
    treated_salary = stratified_avg_salary[  
        (stratified_avg_salary["academic_performance"] == z)  
        & (stratified_avg_salary["internship"] == 1)  
    ]["starting_salary"].values[0]  
    untreated_salary = stratified_avg_salary[  
        (stratified_avg_salary["academic_performance"] == z)  
        & (stratified_avg_salary["internship"] == 0)  
    ]["starting_salary"].values[0]  
    ate_adjusted += p_z * (treated_salary - untreated_salary)
```

Question 3.3:

```
print(f"Adjusted Average Treatment Effect estimate: {ate_adjusted:.4f}")  
print(f"Naive Average Treatment Effect estimate: {naive_ATE:.4f}")
```

Adjusted Average Treatment Effect estimate: 5012.2760

Naive Average Treatment Effect estimate: 9204.7018

New conclusion: completing an internship increases the annual starting salary by 5013 US dollars, on average.

This is much lower than the originally estimated value 9204.

The true ATE was indeed 5000 (differences are due to statistical noise), and so the original estimate would have severely overestimated the true causal effect.

This notebook was converted with `convert.ploomber.io`