Assignment 3

Push Notifications for a Mobile Fitness App

Zaheer Abbas

2025-06-11

Table of contents

Loading data set

```
print(dataset.head(5),dataset.columns)
```

```
   age  gender region  sessions_per_day  past_engagement device_type  \
0   56    Male     US                 2        37.331264         iOS
1   46  Female     US                 3        45.346106     Android
2   32  Female     US                 1       111.716016         iOS
3   60  Female     US                 7         6.564358         iOS
4   25  Female     US                 8         4.225939         iOS

   app_version  notification_sent  pop_up_rem  notification_opened  engagement
0            1                  0           1                    1   22.915902
1            3                  0           0                    1   20.043720
2            2                  0           0                    0   28.373763
3            1                  0           0                    0   23.419303
4            2                  1           1                    0   16.772125   Index(['age', 'gende
       'device_type', 'app_version', 'notification_sent', 'pop_up_rem',
       'notification_opened', 'engagement'],
      dtype='object')
```

1 Conditional outcome regression

1.0.1 Regress engagement on notification_sent as well as covariates.

```
import statsmodels.formula.api as smf
# Ols regression model
model=smf.ols(formula="""engagement~notification_sent+age + gender + region + device_type
+ app_version + sessions_per_day
+ past_engagement """, data=dataset).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:             engagement   R-squared:                       0.484
Model:                            OLS   Adj. R-squared:                  0.483
Method:                 Least Squares   F-statistic:                     1040.
Date:                Wed, 11 Jun 2025   Prob (F-statistic):               0.00
Time:                        22:39:54   Log-Likelihood:                -33413.
No. Observations:               10000   AIC:                         6.685e+04
Df Residuals:                    9990   BIC:                         6.692e+04
Df Model:                           9
Covariance Type:            nonrobust
======================================================================================
                         coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------
Intercept             12.9202      0.373     34.656      0.000      12.189      13.651
gender[T.Male]         0.1453      0.137      1.062      0.288      -0.123       0.414
region[T.EU]          -0.0852      0.185     -0.459      0.646      -0.449       0.278
region[T.US]          -0.0976      0.187     -0.523      0.601      -0.464       0.268
device_type[T.iOS]    -0.0386      0.140     -0.276      0.783      -0.313       0.236
notification_sent      7.0433      0.153     46.142      0.000       6.744       7.342
age                   -0.0086      0.006     -1.540      0.124      -0.019       0.002
app_version            0.1208      0.097      1.240      0.215      -0.070       0.312
sessions_per_day       0.5379      0.040     13.460      0.000       0.460       0.616
past_engagement        0.1033      0.002     59.730      0.000       0.100       0.107
==============================================================================
Omnibus:                       12.983   Durbin-Watson:                   1.981
Prob(Omnibus):                  0.002   Jarque-Bera (JB):               11.965
Skew:                           0.050   Prob(JB):                      0.00252
Kurtosis:                       2.863   Cond. No.                         466.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Interpretation of Results and Conditional Average Treatment Effect

The regression results indicate that sending a notification (notification_sent) has a significant positive effect on user engagement.

- Here are only the covariates with a strong (statistically significant) impact on engagement from regression:
- notification_sent: Increases engagement by about 7.03 minutes (highly significant)
- sessions_per_day: Each additional session increases engagement by about 0.54 minutes (significant)
- past_engagement: Each additional minute of past engagement increases current engagement by about 0.10 minutes (very significant)

Receiving a notification is associated with an average increase of approximately *7.04 minutes of app engagement*. this number is called the Conditional Average Treatment Effect (CATE) because it shows the effect of the notification while considering these other factors. Other things like the number of sessions per day, past engagement, but demographic factors didn't have a significant impact in this model.

2 Propensity Score Matching

2.0.1 Estimate propensity scores for *notification_sent*

```
from sklearn.linear_model import LogisticRegression
dataset_numeric = pd.get_dummies(dataset, drop_first=True)
X = dataset_numeric.drop('notification_sent', axis=1)
y = dataset_numeric['notification_sent']

# Fit propensity score model
ps_model = LogisticRegression(max_iter=10000)
ps_model.fit(X, y)

# Add propensity scores to dataset_numeric
dataset_numeric['propensity_score'] = ps_model.predict_proba(X)[:, 1]
print(dataset_numeric['propensity_score'])
```

```
0       0.274057
1       0.235229
2       0.613432
3       0.349250
4       0.138153
          ...
9995    0.032487
9996    0.116402
9997    0.237166
9998    0.037332
9999    0.155098
Name: propensity_score, Length: 10000, dtype: float64
```

2.0.2 Nearest neighbor matching

```
ATT (1:1 Matching): 0.1320
ATT (1:3 matching): 0.1554
```

Results: Both values are positive, meaning the notification increased engagement.

2.0.3 covariate balance before and after matching

```
 Covariate Balance Before Matching:

 Covariate Balance After 1:3 Matching:
```

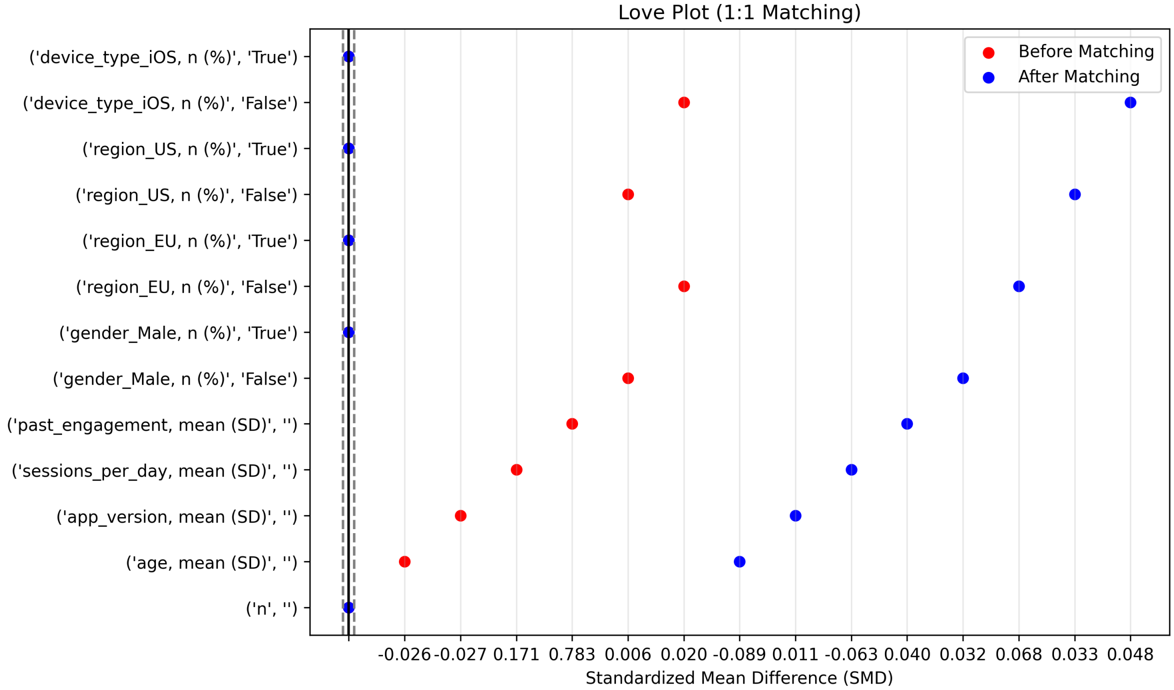
```
print("\n Covariate Balance Before Matching:")
print(table_before.tabulate(tablefmt="github"))

print("\n Covariate Balance  Matching:")
print(table_after_1to1.tabulate(tablefmt="github"))
print(table_after_1to3.tabulate(tablefmt="github"))
```

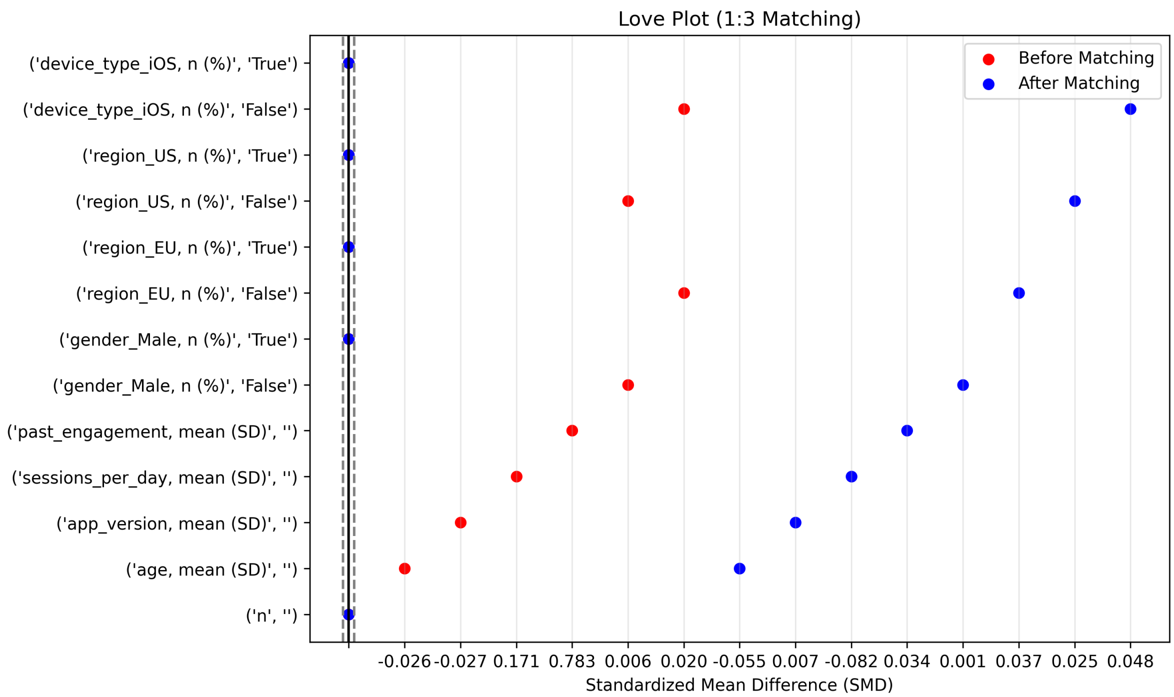
```
 Covariate Balance Before Matching:
|                              |       | Missing   | Overall     | 0           | 1           | SMD (0,
|------------------------------|-------|-----------|-------------|-------------|-------------|--------
| n                            |       |           | 10000       | 6229        | 3771        |
| age, mean (SD)               |       | 0         | 39.0 (12.3) | 39.1 (12.3) | 38.8 (12.2) | -0.026
| app_version, mean (SD)       |       | 0         | 1.9 (0.7)   | 1.9 (0.7)   | 1.9 (0.7)   | -0.027
| sessions_per_day, mean (SD)  |       | 0         | 3.0 (1.7)   | 2.9 (1.7)   | 3.2 (1.8)   | 0.171
| past_engagement, mean (SD)   |       | 0         | 60.6 (42.6) | 48.2 (31.7) | 81.0 (49.9) | 0.783
| gender_Male, n (%)           | False |           | 5015 (50.1) | 3117 (50.0) | 1898 (50.3) | 0.006
|                              | True  |           | 4985 (49.9) | 3112 (50.0) | 1873 (49.7) |
| region_EU, n (%)             | False |           | 5947 (59.5) | 3681 (59.1) | 2266 (60.1) | 0.020
|                              | True  |           | 4053 (40.5) | 2548 (40.9) | 1505 (39.9) |
| region_US, n (%)             | False |           | 6102 (61.0) | 3808 (61.1) | 2294 (60.8) | 0.006
|                              | True  |           | 3898 (39.0) | 2421 (38.9) | 1477 (39.2) |
| device_type_iOS, n (%)       | False |           | 6037 (60.4) | 3737 (60.0) | 2300 (61.0) | 0.020
|                              | True  |           | 3963 (39.6) | 2492 (40.0) | 1471 (39.0) |

 Covariate Balance  Matching:
|                              |       | Missing   | Overall     | 0           | 1           | SMD (0,
|------------------------------|-------|-----------|-------------|-------------|-------------|--------
| n                            |       |           | 7542        | 3771        | 3771        |
| age, mean (SD)               |       | 0         | 39.4 (12.3) | 39.9 (12.4) | 38.8 (12.2) | -0.089
| app_version, mean (SD)       |       | 0         | 1.9 (0.7)   | 1.9 (0.7)   | 1.9 (0.7)   | 0.011
| sessions_per_day, mean (SD)  |       | 0         | 3.2 (1.8)   | 3.3 (1.9)   | 3.2 (1.8)   | -0.063
| past_engagement, mean (SD)   |       | 0         | 80.0 (48.4) | 79.1 (46.9) | 81.0 (49.9) | 0.040
```

gender_Male, n (%)	False		3736 (49.5)	1838 (48.7)	1898 (50.3)	0.032
	True		3806 (50.5)	1933 (51.3)	1873 (49.7)	
region_EU, n (%)	False		4406 (58.4)	2140 (56.7)	2266 (60.1)	0.068
	True		3136 (41.6)	1631 (43.3)	1505 (39.9)	
region_US, n (%)	False		4649 (61.6)	2355 (62.5)	2294 (60.8)	0.033
	True		2893 (38.4)	1416 (37.5)	1477 (39.2)	
device_type_iOS, n (%)	False		4687 (62.1)	2387 (63.3)	2300 (61.0)	0.048
	True		2855 (37.9)	1384 (36.7)	1471 (39.0)	

		Missing	Overall	0	1	SMD (0,
n			15084	11313	3771	
age, mean (SD)		0	39.3 (12.3)	39.5 (12.3)	38.8 (12.2)	-0.055
app_version, mean (SD)		0	1.9 (0.7)	1.9 (0.7)	1.9 (0.7)	0.007
sessions_per_day, mean (SD)		0	3.3 (1.9)	3.3 (1.9)	3.2 (1.8)	-0.082
past_engagement, mean (SD)		0	79.7 (48.7)	79.3 (48.3)	81.0 (49.9)	0.034
gender_Male, n (%)	False		7597 (50.4)	5699 (50.4)	1898 (50.3)	0.001
	True		7487 (49.6)	5614 (49.6)	1873 (49.7)	
region_EU, n (%)	False		8856 (58.7)	6590 (58.3)	2266 (60.1)	0.037
	True		6228 (41.3)	4723 (41.7)	1505 (39.9)	
region_US, n (%)	False		9315 (61.8)	7021 (62.1)	2294 (60.8)	0.025
	True		5769 (38.2)	4292 (37.9)	1477 (39.2)	
device_type_iOS, n (%)	False		9464 (62.7)	7164 (63.3)	2300 (61.0)	0.048
	True		5620 (37.3)	4149 (36.7)	1471 (39.0)	

2.0.4 love polt





Results - *Before matching*, notable imbalance exists between the treated and untreated groups on key behavioral covariates. In particular, past_engagement shows a large standardized mean difference (SMD = 0.783), indicating that users who received notifications were already much more active than those who did not. Similarly, sessions_per_day has an SMD of 0.171, also exceeding the common threshold of 0.1, suggesting further imbalance. Other covariates such as age, app_version, and demographic variables show good balance with low SMDs and non-significant p-values. Overall, these results highlight the need for matching to reduce bias and improve comparability between groups before estimating the treatment effect.

- *After 1:1 matching*, covariate balance improved notably. The large imbalance in past_engagement decreased from 0.783 to 0.040, indicating better comparability. Other key variables like sessions_per_day and age also showed reduced differences with SMDs below 0.1. Most covariates now have minimal differences, though slight imbalance remains in region_EU and device_type_iOS. Overall, matching made the treated and untreated groups much more similar, strengthening the validity of treatment effect analysis.
- *After 1:3 matching*, covariate balance further improved compared to before matching. The standardized mean differences for key variables like age, sessions_per_day, and past_engagement are all below 0.1, indicating good balance. Differences in categorical variables such as gender_Male, region_EU, and device_type_iOS are minimal, with most p-values showing no significant difference. Overall, 1:3 matching effectively reduced imbalance between treated and untreated groups, enhancing the reliability of subsequent treatment effect estimates.

2.0.5 ATT and ATE

As i already calculated ATT for 1 to 1 and 1 to3 let just call 1 to 1 here only.

```
print("--1 to 1 Matching Results-- ")
print(f" \n ATT: {att1:.4f}")
print(f"Standard Error: {se_att:.4f}")
print(f"t-statistic: {t_stat:.4f}")
print(f"p-value: {p_value:.4f}")
```

```
--1 to 1 Matching Results--

 ATT: 0.1320
Standard Error: 0.0346
t-statistic: 3.8196
p-value: 0.0001
```

The estimated Average Treatment Effect on the Treated (ATT) is 0.132, indicating that the treatment increases the outcome by approximately 0.13 units for those who received it. The result is statistically significant, supported by a small standard error of 0.035, a t-statistic of 3.82, and a p-value of 0.0001. This suggests strong evidence that the treatment has a meaningful positive impact on the treated group.

3 Inverse Probability Weighting

3.0.1 stablized weights calculation.

```
# Check the first few weights
print(dataset_numeric[['notification_sent', 'propensity_score', 'stabilized_weight']].head(15))#
```

```
    notification_sent  propensity_score  stabilized_weight
0                   0          0.274057           0.858056
1                   0          0.235229           0.814492
2                   0          0.613432           1.611361
3                   0          0.349250           0.957204
4                   1          0.138153           2.729573
5                   0          0.083874           0.679929
6                   0          0.585990           1.504552
7                   0          0.289347           0.876518
8                   0          0.146303           0.729650
9                   1          0.281525           1.339492
10                  0          0.074528           0.673062
11                  0          0.050954           0.656343
12                  1          0.374746           1.006281
13                  0          0.346772           0.953573
14                  1          0.526457           0.716297
```

3.0.2 Covariate balance under the weights

```
MultiIndex([(                        'n',      ''),
            (             'age, mean (SD)',      ''),
            (     'app_version, mean (SD)',      ''),
            ('sessions_per_day, mean (SD)',      ''),
            ( 'past_engagement, mean (SD)',      ''),
            (         'gender_Male, n (%)', 'False'),
            (         'gender_Male, n (%)',  'True'),
            (           'region_EU, n (%)', 'False'),
            (           'region_EU, n (%)',  'True'),
            (           'region_US, n (%)', 'False'),
            (           'region_US, n (%)',  'True'),
            (     'device_type_iOS, n (%)', 'False'),
            (     'device_type_iOS, n (%)',  'True')],
           )
Weighted Balance Table:
```

```
print(table_weighted)
```

		Grouped by notification_sent		
		Missing	Overall	
n			10000	622
age, n (%)	18		224 (2.2)	145 (2.3
	19		238 (2.4)	156 (2.5
	20		228 (2.3)	142 (2.3
	21		235 (2.4)	146 (2.3
	22		216 (2.2)	127 (2.0
	23		226 (2.3)	128 (2.1
	24		201 (2.0)	121 (1.9
	25		230 (2.3)	144 (2.3
	26		229 (2.3)	125 (2.0
	27		223 (2.2)	149 (2.4
	28		237 (2.4)	154 (2.5
	29		240 (2.4)	145 (2.3
	30		249 (2.5)	154 (2.5
	31		221 (2.2)	137 (2.2
	32		240 (2.4)	143 (2.3
	33		231 (2.3)	147 (2.4
	34		260 (2.6)	163 (2.6
	35		245 (2.5)	149 (2.4
	36		231 (2.3)	142 (2.3
	37		230 (2.3)	135 (2.2
	38		244 (2.4)	151 (2.4
	39		239 (2.4)	156 (2.5
	40		252 (2.5)	160 (2.6
	41		237 (2.4)	156 (2.5
	42		244 (2.4)	151 (2.4
	43		267 (2.7)	161 (2.6
	44		195 (1.9)	106 (1.7
	45		255 (2.5)	158 (2.5
	46		242 (2.4)	162 (2.6
	47		225 (2.2)	133 (2.1
	48		203 (2.0)	130 (2.1
	49		237 (2.4)	143 (2.3
	50		232 (2.3)	148 (2.4
	51		229 (2.3)	145 (2.3
	52		242 (2.4)	160 (2.6
	53		227 (2.3)	141 (2.3
	54		241 (2.4)	160 (2.6
	55		225 (2.2)	135 (2.2
	56		244 (2.4)	158 (2.5
	57		223 (2.2)	150 (2.4
	58		213 (2.1)	133 (2.1
	59		239 (2.4)	155 (2.5
	60		211 (2.1)	125 (2.0
app_version, n (%)	1		2999 (30.0)	1834 (29.4
	2		4973 (49.7)	3121 (50.1
	3		2028 (20.3)	1274 (20.5

```
sessions_per_day, n (%)        0                                          477 (4.8)        333 (5.3
                               1                                         1525 (15.2)      1022 (16.4
                               10                                          17 (0.2)         10 (0.2
                               11                                           1 (0.0)          1 (0.0
                               13                                           1 (0.0)          0 (0.0
                               2                                         2227 (22.3)      1423 (22.8
                               3                                         2276 (22.8)      1433 (23.0
                               4                                         1648 (16.5)       979 (15.7
                               5                                         1027 (10.3)       603 (9.7
                               6                                          522 (5.2)        286 (4.6
                               7                                          198 (2.0)        105 (1.7
                               8                                           58 (0.6)         24 (0.4
                               9                                           23 (0.2)         10 (0.2
past_engagement, median [Q1,Q3]                              0  51.2 [29.2,82.1]  42.2 [24.5,65.3
gender_Male, n (%)             False                                     5015 (50.1)      3117 (50.0
                               True                                      4985 (49.9)      3112 (50.0
region_EU, n (%)               False                                     5947 (59.5)      3681 (59.1
                               True                                      4053 (40.5)      2548 (40.9
region_US, n (%)               False                                     6102 (61.0)      3808 (61.1
                               True                                      3898 (39.0)      2421 (38.9
device_type_iOS, n (%)         False                                     6037 (60.4)      3737 (60.0
                               True                                      3963 (39.6)      2492 (40.0
```

3.0.3 weighted regression

weighted regression of engagement on notification_sent.

```python
# Fit the weighted least squares regression
model = sm.WLS(y, X, weights=weights).fit()

print(model.summary())
```

```
                            WLS Regression Results
==============================================================================
Dep. Variable:             engagement   R-squared:                       0.000
Model:                            WLS   Adj. R-squared:                 -0.000
Method:                 Least Squares   F-statistic:                    0.3490
Date:                Wed, 11 Jun 2025   Prob (F-statistic):              0.555
Time:                        22:40:01   Log-Likelihood:                -37382.
No. Observations:               10000   AIC:                         7.477e+04
Df Residuals:                    9998   BIC:                         7.478e+04
Df Model:                           1
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
const                23.0835      0.119    194.319      0.000      22.851      23.316
```

```
notification_sent     0.1137      0.193      0.591      0.555      -0.264       0.491
==============================================================================
Omnibus:                     2293.674   Durbin-Watson:                   1.981
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            62264.092
Skew:                           0.482   Prob(JB):                         0.00
Kurtosis:                      15.186   Cond. No.                         2.43
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

4 Double-robust Estimation.

Estimate both outcome and propensity models and obtain predictions.
Manually compute the doubly-robust average treatment effect using these predictions.

```python
print("Doubly Robust ATE:", dr_ate_value)
print("Predicted outcome if treated (y1):\n", y1_pred.head())
print("Predicted outcome if control (y0):\n", y0_pred.head())
```

```
Doubly Robust ATE: 0.09928391405354375
Predicted outcome if treated (y1):
 0    29.920888
1    29.920888
2    29.920888
3    29.920888
4    29.920888
dtype: float64
Predicted outcome if control (y0):
 0    19.333025
1    19.333025
2    19.333025
3    19.333025
4    19.333025
dtype: float64
```

```python
#  Output Results

print("Bootstrapped Doubly Robust ATE Estimate:", result["ate"])
print("Standard Error:", result["std_error"])
print("T-statistic:", result["t_stat"])
print("P-value:", result["p_value"])
```

```
Bootstrapped Doubly Robust ATE Estimate: 7.114993901664724
Standard Error: 0.1591355597781598
T-statistic: 44.71027036058603
P-value: 0.0
```

The Bootstrapped Doubly Robust Average Treatment Effect (ATE) estimate is 7.11, suggesting that the treatment (e.g., sending a notification) increases the outcome variable by approximately 7.11 units on average. This estimate is highly precise, with a standard error of 0.159, leading to a very large t-statistic of 44.71 and a p-value of 0.0. These values indicate that the treatment effect is statistically significant at conventional levels, providing strong evidence that the treatment has a meaningful and positive impact on the outcome.

5 Sensitivity Analysis to Unobserved Confounding

```
print(f"Robustness Value (RV): {rv:.4f}")
print(f"Strongest observed covariate: {strongest_covariate} (Partial R² = {strongest_partial_r2:.4f})
```

```
Robustness Value (RV): 482.6835
Strongest observed covariate: past_engagement (Partial R² = 0.3616)

 ATE estimate is robust: an unobserved confounder would need to be stronger than any observed covaria
```

6 Instrumental-variable Estimation

Local Average Treatment Effect (LATE) manually :

```
Z = dataset['pop_up_rem']
D = dataset['notification_sent']
Y = dataset['engagement']

# Compute means
EY_Z1 = np.mean(Y[Z == 1])
EY_Z0 = np.mean(Y[Z == 0])

ED_Z1 = np.mean(D[Z == 1])
ED_Z0 = np.mean(D[Z == 0])

# Compute LATE
LATE = (EY_Z1 - EY_Z0) / (ED_Z1 - ED_Z0)

print(f"Local Average Treatment Effect (LATE): {LATE}")
```

```
Local Average Treatment Effect (LATE): 120.90999527002054
```

6.0.1 Two-stage least squares (2SLS) regressions

```python
print("     -------- without covariates -------- ")

print(iv_no_cov.summary)

print("\n       ------ with covariates  ------- ")

print(iv_with_cov.summary)
```

```
     -------- without covariates --------
                          IV-2SLS Estimation Summary
==============================================================================
Dep. Variable:             engagement   R-squared:                      0.3408
Estimator:                    IV-2SLS   Adj. R-squared:                 0.3407
No. Observations:               10000   F-statistic:                    133.39
Date:                Wed, Jun 11 2025   P-value (F-stat)                0.0000
Time:                        22:40:32   Distribution:                  chi2(1)
Cov. Estimator:                robust

                                 Parameter Estimates
=======================================================================================
                     Parameter  Std. Err.     T-stat    P-value    Lower CI    Upper CI
---------------------------------------------------------------------------------------
Intercept               18.792     0.3980     47.212     0.0000      18.012      19.572
notification_opened     9.3457     0.8092     11.550     0.0000      7.7597      10.932
=======================================================================================

Endogenous: notification_opened
Instruments: pop_up_rem
Robust Covariance (Heteroskedastic)
Debiased: False

       ------ with covariates  -------
                          IV-2SLS Estimation Summary
==============================================================================
Dep. Variable:             engagement   R-squared:                      0.6374
Estimator:                    IV-2SLS   Adj. R-squared:                 0.6372
No. Observations:               10000   F-statistic:                 1.001e+04
Date:                Wed, Jun 11 2025   P-value (F-stat)                0.0000
Time:                        22:40:32   Distribution:                  chi2(7)
Cov. Estimator:                robust

                                 Parameter Estimates
=======================================================================================
                     Parameter  Std. Err.     T-stat    P-value    Lower CI    Upper CI
---------------------------------------------------------------------------------------
Intercept               10.162     0.2919     34.815     0.0000      9.5901      10.734
sessions_per_day        0.6024     0.0337     17.877     0.0000      0.5364      0.6685
```

```
past_engagement          0.1209     0.0016     74.073     0.0000      0.1177      0.1241
gender_Male              0.0609     0.1147     0.5305     0.5957     -0.1640      0.2857
region_EU               -0.1423     0.1566    -0.9085     0.3636     -0.4492      0.1647
region_US               -0.1343     0.1580    -0.8503     0.3952     -0.4439      0.1753
device_type_iOS         -0.0545     0.1169    -0.4662     0.6411     -0.2837      0.1747
notification_opened      8.5339     0.6056     14.091     0.0000      7.3469      9.7209
==========================================================================================

Endogenous: notification_opened
Instruments: pop_up_rem
Robust Covariance (Heteroskedastic)
Debiased: False
```

Short description of the above two-stage least squares (2SLS) regressions In the model without covariates, the estimated effect of notification_opened on engagement is 9.35, with a standard error of 0.81. This effect is statistically significant ($p < 0.001$), and the 95% confidence interval ranges from 7.76 to 10.93. The model explains 34.1% of the variation in engagement, as indicated by the R-squared value.

In contrast, the model with covariates (including user behavior and demographics like sessions_per_day, past_engagement, gender, region, and device type) estimates a slightly smaller but still highly significant effect of 8.53 for notification_opened (SE = 0.61, $p < 0.001$). The inclusion of covariates improves the precision of the estimate and controls for potential confounding. Notably, past_engagement and sessions_per_day also show strong positive associations with engagement, while demographic variables (e.g., gender, region, device type) are not statistically significant.

Overall, both models consistently show a strong and statistically significant positive impact of opening a notification on engagement

7 Executive Summary

In this Assignment, we looked at whether sending mobile app notifications makes users more engaged. We used data from 10,000 users and focused on two main questions:

- Does getting a notification lead to higher engagement?
- Does actually opening the notification lead to even more engagement?

we employed various causal inference methods including regression adjustment, propensity score matching (PSM), inverse probability weighting (IPW), doubly robust (DR) estimation, and instrumental variables (IV). Sensitivity analyses were also conducted to evaluate robustness against potential unobserved confounding.

Before matching, there was notable imbalance between treated and untreated groups on key covariates—most notably past engagement (SMD = 0.783) and sessions per day (SMD = 0.171)—suggesting the need for adjustment. After applying 1:1 and 1:3 matching, covariate balance improved substantially, with all key variables showing standardized mean differences below 0.1, enhancing the reliability of treatment effect estimates.

Across all methods, we found that sending a notification significantly increased user engagement, with the doubly robust average treatment effect (ATE) estimated at +7.1 points ($p < 0.001$). The effect of opening a notification was even stronger, with IV estimation showing an effect of +9.3 points, indicating that encouraging users to open notifications could yield further gains.

Covariate balance checks and sensitivity metrics suggest our results are robust to observed confounding and reasonably robust to unmeasured factors. These findings support investing in both targeted notifications and designs that increase open rates, as both interventions can meaningfully increase user activity.