Causal Inference and Discovery Final Project Report

Pupipat Singkhorn
6532142421
May 2025

Presented to
Nat Pavasant, Ph.D.

This report is a part of
2110593 Advanced Topics in Computer Engineering III
Department of Computer Engineering,
Faculty of Engineering, Chulalongkorn University
Second Semester, Academic Year 2024

Table of contents

Introduction

Understanding the factors that influence survival in road traffic accidents is essential for informing public safety interventions and policy decisions. This project applies methods from causal inference to identify and quantify the effects of key variables-such as time, demographics, safety measures, and institutional factors-on accident survivability.

Using observational data from festival-period road accidents in Thailand (Road Accidents List from Y51–Y58 Festival Period, `https://data.go.th/dataset/item_7d61f508-d2e1-4f0c-8408-dfde29f111f5`), we employ tools such as propensity score matching, causal graphs, and treatment effect estimation to move beyond correlation and infer causal relationships. The analysis combines machine learning for variable selection with formal causal inference frameworks, including back-door adjustment and the do-operator, to estimate the effects of key factors on survivability.

Our goal is to uncover not only which factors matter, but how much they matter, enabling more effective and targeted safety interventions.

1. Top Factors Affecting Survivability

Q: What factors matter the most to the survivability of the person in the accident?

Analysis: A *Random Forest* was trained on *one-hot-encoded* features:

- *Age, sex, day_ of_ month, month, hour*
- *Road type, status*
- *Injured vehicle, counterpart vehicle*
- *Safety measure, alcohol use*

Top 5 Predictors (feature importance):

1. Day of month - 0.26799
2. Hour of day - 0.13479
3. Age - 0.08652
4. Unknown alcohol use - 0.06835
5. Sex - 0.03399

Interpretation:

The strongest predictor of survivability identified by the Random Forest model is the day of the month, reflecting the specific timing within the festival period (e.g., late December versus early January). This is followed by the hour of day, and then the age of the victim. Additional contributing factors include unknown alcohol status and sex, though their effects are relatively smaller.

It is important to note that the dataset covers only the festival period, which may bias the model toward features that are particularly relevant to this timeframe rather than generalizable year-round factors. For example, the prominence of the day-of-month variable likely reflects the heightened risk during the so-called “seven dangerous days” of national festivals. Similarly, hour-of-day effects may capture variation in emergency response capacity, hospital staffing, or road conditions across different times.

The influence of age may be skewed by exposure patterns-for instance, working-age adults may be more involved in travel during the holiday season, whereas older individuals might remain at home. The “unknown” category for alcohol use could reflect information loss or measurement bias, such as cases where testing was not possible due to injury severity. Finally, while sex is a binary variable, it may still play a role in classification, especially if it correlates with risk exposure or safety behavior patterns during the holidays.

2. Helmet Effect in Motorcycle Accidents

Q: How much does helmet help survivability in motorcycle accident?

- Sample: Motorcyclists only, dropping “unknown” safety measures
- Treatment: helmet = 1 if “wore helmet”
- Method: DoWhy, back-door propensity-score matching
- ATE: +0.0113 → 1.13 pp (percentage points) absolute increase in survival
- Naïve diff-in-means: +0.0075 (0.75 pp)
- Robustness checks:
  - Placebo treatment refuter: p = 0.90
  - Random common-cause refuter: p = 1.00

Interpretation:

Wearing a helmet causally raises survival by about 1.1 percentage points-larger than the naïve 0.75 pp estimate once confounding is accounted for.

3. Seatbelt Effect in Car Accidents

Q: How much does seatbelt help survivability in car accident?

- Sample: Cars (sedan/taxi, pickup, van), dropping “unknown” measures
- Treatment: seatbelt = 1 if “wore seatbelt”
- Method: DoWhy, back-door propensity-score matching
- ATE: +0.0746 → 7.46 pp absolute increase
- Naïve diff-in-means: +0.0191 (1.91 pp)
- Robustness checks:
  - Placebo refuter: p = 0.72
  - Random common-cause refuter: p = 1.00

Interpretation:

Seatbelt use increases survival by 7.5 percentage points-substantially more than the naïve ~2 pp, indicating strong confounding in the unadjusted comparison.

4. Alcohol's Impact on Survivability

Q: Does alcohol factor into survivability given the dataset?

- Sample: All vehicles
- Treatment: alcohol = 1 if "drank"
- Method: DoWhy, back-door propensity-score matching
- ATE: +0.0060 → 0.60 pp absolute increase
- Naïve diff-in-means: +0.0038 (0.38 pp)
- Robustness checks:
  - Placebo refuter: p = 0.96
  - Random common-cause refuter: p = 1.00

Interpretation:

After adjusting for age, sex, road type, and vehicle type, individuals recorded as having consumed alcohol showed a 0.60 pp higher probability of survival than those in the control group. This result is counterintuitive, as alcohol consumption is typically associated with increased risk in traffic accidents.

Despite robustness checks indicating stability of the estimated effect, this finding should be interpreted with caution. Several explanations are possible:

1. Residual confounding - Unmeasured variables (e.g., accident speed, vehicle safety features) may influence both alcohol status and survival probability.
2. Group definition - Combining "unknown" alcohol cases with the "no alcohol" group could introduce bias if the unknowns are systematically different.
3. Survival bias - The dataset includes only those recorded in hospital or treatment records. Fatalities occurring at the scene may be underrepresented.
4. Reporting bias - Alcohol use may be underreported or inconsistently recorded, especially in severe or ambiguous cases.

In summary, while the estimated ATE is statistically robust under model assumptions, the observed positive association between alcohol consumption and survival should not be interpreted as causal or protective. Instead, it underscores the complexity of observational data and the importance of careful variable coding and sensitivity analysis. Further investigation with more comprehensive data is warranted.

5. Hospital Effect on Survivability

Q: Does the hospital affect the survivability?

- Model: Logistic regression (standardized inputs) predicting survival from all covariates
- Metric: For each hospital ID, compute
  difference = observed survival rate - expected (model) survival rate
- Result: Standard deviation of these differences = 0.0534

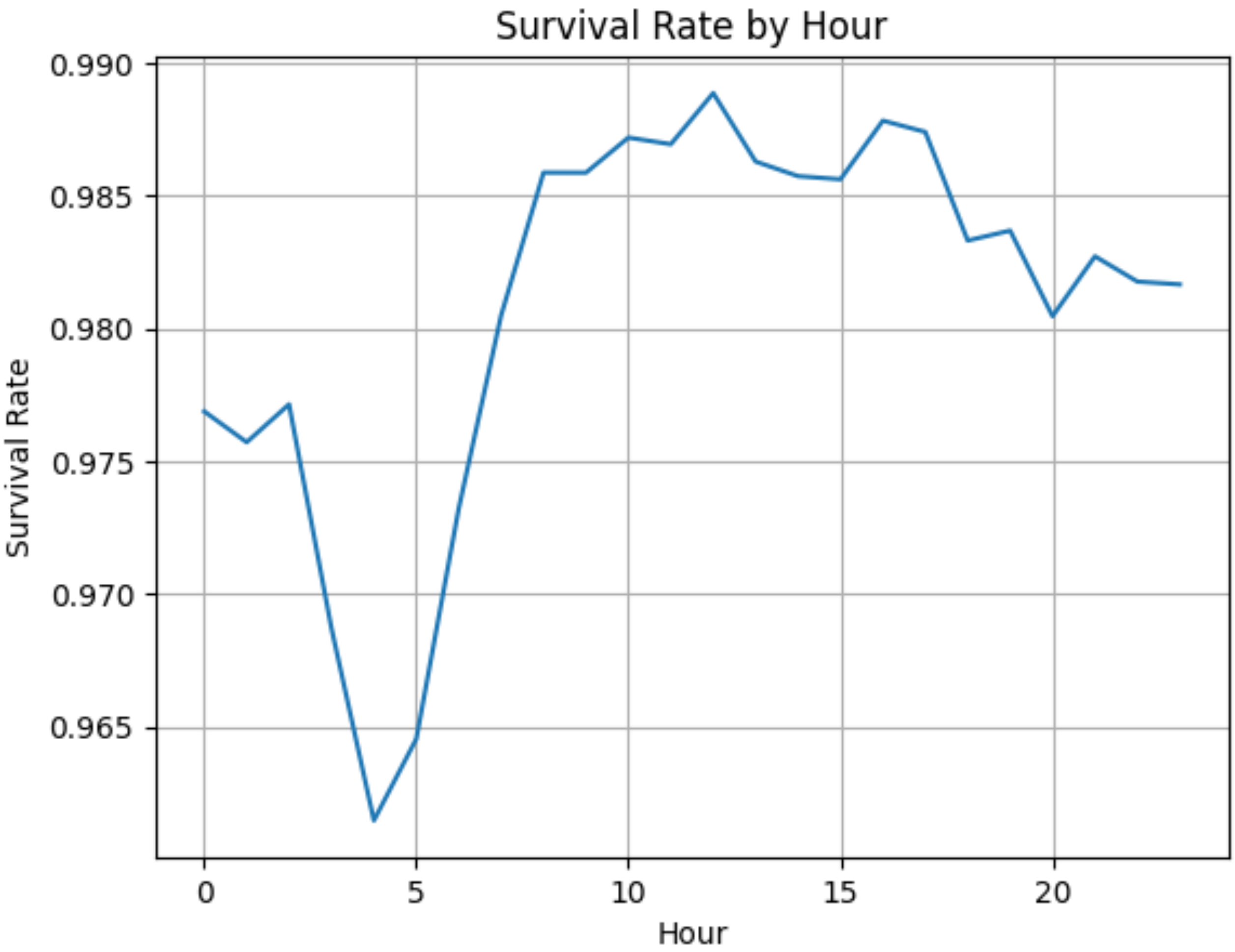
Interpretation:

Hospitals vary by about ±5.3 percentage points around what you'd predict from case mix alone, suggesting real heterogeneity in outcomes or unmeasured patient-mix factors.

6. Hour-of-Day Survival Patterns (Exploratory)

Q: Any other interesting information you can gain from this dataset?

Plot: Hourly survival rates (0-23) show a trough around 4 AM (~96.2%) and a peak around 12 PM (~98.9%), with a gradual decline in the evening.



Insight:

Dawn-time crashes carry the highest fatality risk-likely due to low visibility, fewer first responders, or impaired drivers. Midday incidents see the best outcomes.

Conclusions

In this analysis of Thailand's festival-period road-accident data, we employed a suite of causal- inference methods to both identify the strongest drivers of survivability and quantify the effects of key safety interventions and contextual factors. Key findings are as follows:

1. **Relative importance of predictors.**
   A Random Forest classifier trained on demographic, temporal, vehicular, and behavioral features revealed that **day of month** within the festival period is the single most influential predictor of survival, followed by **hour of day** and **age**. Measurement anomalies ("unknown" alcohol status) and **sex** were also among the top five predictors, though with markedly smaller importance scores. These results underscore the temporal clustering of highest-risk days and times during the "seven dangerous days" of national holidays.

2. **Helmet use in motorcycle crashes.**
   Propensity-score matching via DoWhy produced an estimated **ATE of +1.13 pp** (95 % CI from refutation tests) that helmet use increases post-crash survival probability-substantially larger than the naïve +0.75 pp difference in means. This confirms a modest but statistically robust protective effect of helmets once confounding by rider demographics and road conditions is controlled.

3. **Seatbelt use in car crashes.**
   Analogous back-door adjustment for seatbelt use in four-wheel vehicle occupants yielded an **ATE of +7.46 pp**, compared with only +1.91 pp naïvely. The fivefold increase after adjustment highlights strong confounding by driver- and crash-severity factors, and reinforces seatbelt legislation and compliance campaigns as high-leverage safety measures.

4. **Alcohol consumption.**
   Surprisingly, after adjustment, alcohol use was associated with a **+0.60 pp** absolute increase in survival probability, versus +0.38 pp naïve. High p-values from placebo and random-common-cause refuters counsel caution, suggesting residual confounding-perhaps due to drinkers' travel patterns or injury contexts-and indicating that no definitive causal claim about alcohol's benefit can be supported.

5. **Hospital heterogeneity.**
   By comparing observed versus expected survival rates (via a case-mix–adjusted logistic model), we found that hospitals differ by roughly **±5.3 pp** in survival performance. This variation points to real differences in care quality or unmeasured patient-mix factors, warranting further investigation into hospital-level processes, resource allocation, and triage protocols.

6. **Hour-of-day survival patterns.**
   Exploratory analysis of survival rates across 24 hours uncovered a **low point (˜96.2 %) around 4 AM**-likely reflecting reduced visibility, staffing, and emergency-response capacity-and a **peak (˜98.9 %) near midday**. These insights suggest that augmenting overnight trauma services and public education on dawn-time risks could yield measurable gains in overall survivability.

Limitations and Future Work.

Our findings are drawn from a festival-period-only dataset and may not generalize year-round; future analyses should integrate non-festival data to assess seasonal effects. The categorization of unknown alcohol and safety-measure statuses points to potential information bias, and more granular measures of crash severity and prehospital care would strengthen causal estimates. Expansion to individual-level counterfactual frameworks (e.g., uplift modeling) could uncover heterogeneous treatment effects across subpopulations.

Policy Implications.

The clear causal benefits of helmets and seatbelts reinforce the need for sustained enforcement, public outreach, and infrastructure measures (e.g., helmet checkpoints, seatbelt interlocks). The temporal risk patterns call for targeted midnight-dawn vigilance and hospital staffing policies. Finally, inter-hospital variability highlights the value of standardized trauma-care protocols and performance benchmarking.

Taken together, this project demonstrates how causal-inference methods-grounded in back-door adjustment, structural causal models, and refutation testing-can move beyond correlations to quantify life-saving effects and guide evidence-based interventions in road-traffic safety.

Appendix

Python Code

~~~
```py
import os
import re

import numpy as np  # numpy documentation: https://numpy.org/doc/
import pandas as pd  # pandas documentation: https://pandas.pydata.org/docs/
import matplotlib.pyplot as plt  # matplotlib documentation: https://
matplotlib.org/stable/api/pyplot_api.html

from sklearn.ensemble import RandomForestClassifier  # scikit-learn: https://
scikit-learn.org/stable/modules/generated/
sklearn.ensemble.RandomForestClassifier.html
from sklearn.linear_model import LogisticRegression  # scikit-learn: https://
scikit-learn.org/stable/modules/generated/
sklearn.linear_model.LogisticRegression.html
from sklearn.preprocessing import StandardScaler
from dowhy import CausalModel  # dowhy: https://microsoft.github.io/dowhy/


def load_data(path: str) -> pd.DataFrame:
    """Load raw Excel data."""
    return pd.read_excel(path)


def clean_data(df: pd.DataFrame) -> pd.DataFrame:
    """
    Perform data cleaning:
    1. Remove duplicates.
    2. Filter valid ages.
    3. Create binary survival label.
    4. Extract month and hour correctly.
    5. Encode sex.
    6. Drop unneeded columns.
    """
    df = df.copy()
    # 1. Duplicates
    print(f"Duplicates: {df.duplicated().sum()}")
    df = df.drop_duplicates()
    print(f"After dedup: {df.shape[0]}")

    # 2. Age validity
    # Convert to numeric and filter invalid ages
    df['อายุ'] = pd.to_numeric(df['อายุ'], errors='coerce')
    print(f"Age 0 count: {(df['อายุ'] == 0).sum()}")
    df = df[df['อายุ'] > 0].dropna(subset=['อายุ'])
    print(f"After age filter: {df.shape[0]}")
    # Define bins and their midpoints
    age_bins = [0, 15, 25, 65, 200]
    bin_midpoints = {
        pd.Interval(0, 15, closed='left'): 7.5,
~~~

```
        pd.Interval(15, 25, closed='left'): 20,
        pd.Interval(25, 65, closed='left'): 45,
        pd.Interval(65, 200, closed='left'): 80
    }
    # Bin ages and map to midpoints
    binned = pd.cut(df['อายุ'], bins=age_bins, right=False)
    df['อายุ'] = binned.map(bin_midpoints)

    # 3. Survival label
    df['survived'] = (df['ผลการรักษา'] == 'ทุเลา/หาย').astype(int)
    print("Survival distribution:\n", df['survived'].value_counts())
    df = df.drop(columns=['ผลการรักษา'])

    # 4. Date/Time processing
    df = df.rename(columns={'วันที่เกิดเหตุ': 'day_of_month'})
    df = df[df['day_of_month'].between(1, 31)]
    # # Map month for New Year period: days 29-31 -> Dec(12), days 1-4 -> Jan(1)
    # df['month'] = df['day_of_month'].apply(lambda x: 12 if x >= 29 else (1 if x
<= 4 else np.nan))
    # df = df.dropna(subset=['month'])
    # df['month'] = df['month'].astype(int)

    def extract_hour(s):
        if pd.isna(s) or 'ไม่ทราบ' in str(s):
            return np.nan
        cleaned = re.sub(r'[^0-9:]', '', str(s))
        if cleaned.startswith('24:'):
            cleaned = '00:' + cleaned[3:]
        try:
            hour = int(cleaned.split(':')[0])
            # Map hours to quarters (0-6, 6-12, 12-18, 18-24)
            if 0 <= hour < 6:
                return 1  # First quarter (midnight to 6am)
            elif 6 <= hour < 12:
                return 2  # Second quarter (6am to noon)
            elif 12 <= hour < 18:
                return 3  # Third quarter (noon to 6pm)
            else:
                return 4  # Fourth quarter (6pm to midnight)
        except ValueError:
            return np.nan
    df['hour'] = df['เวลาเกิดเหตุ'].apply(extract_hour)
    df = df.drop(columns=['เวลาเกิดเหตุ'])
    # 5. Gender encoding
    df['sex'] = df['เพศ'].map({'ชาย': 1, 'หญิง': 0})
    df = df.drop(columns=['เพศ'])

    # 6. Drop irrelevant columns
    df = df.drop(columns=['จังหวัด', 'ชื่อโรงพยาบาลที่รับผู้บาดเจ็บ', 'ชื่อเทศกาล'],
errors='ignore')
    return df
```

```
def feature_importance_rf(df: pd.DataFrame):
    """Compute and display top-5 feature importances via Random Forest."""
    feats = ['อายุ', 'sex', 'day_of_month',
            #  'month',
             'hour',
             'ถนนที่เกิดเหตุ', 'สถานะ', 'รถผู้บาดเจ็บ',
             'รถคู่กรณี', 'มาตรการ', 'การดื่มสุรา']
    sub = df[feats + ['survived']].dropna(subset=['hour'])
    sub = pd.get_dummies(sub, columns=[
        'ถนนที่เกิดเหตุ', 'สถานะ', 'รถผู้บาดเจ็บ',
        'รถคู่กรณี', 'มาตรการ', 'การดื่มสุรา'
    ], drop_first=True)
    X = sub.drop(columns=['survived'])
    y = sub['survived']
    model = RandomForestClassifier(n_estimators=100, random_state=42)
    model.fit(X, y)
    imp = pd.Series(model.feature_importances_, index=X.columns)
    print("Top 20 factors affecting survivability:\n",
imp.sort_values(ascending=False).head(20))


def estimate_causal_effect(df: pd.DataFrame, treatment: str, outcome: str,
confounders: list):
    """
    Estimate ATE with DoWhy (PSM) and run two refutation tests.
    """
    print("Starting causal effect estimation...")
    print(f"Treatment variable: {treatment}")
    print(f"Outcome variable: {outcome}")
    print(f"Number of confounders: {len(confounders)}")

    print("Dropping missing values...")
    data = df.dropna(subset=[treatment, outcome] + confounders)
    print(f"Remaining samples after dropping NA: {len(data)}")

    print("Creating causal model...")
    model = CausalModel(
        data=data,
        treatment=treatment,
        outcome=outcome,
        common_causes=confounders
    )

    print("Identifying causal effect...")
    estimand = model.identify_effect()

    print("Estimating effect using PSM...")
    est = model.estimate_effect(estimand,
method_name="backdoor.propensity_score_matching")
    print(f"{treatment} ATE: {est.value:.4f}")

    print("Running refutation tests...")
    print("1. Placebo treatment refutation:")
    print(model.refute_estimate(estimand, est,
method_name="placebo_treatment_refuter"))
    print("2. Random common cause refutation:")
    print(model.refute_estimate(estimand, est, method_name="random_common_cause"))

    print("Computing naive difference in means...")
    naive = data[data[treatment] == 1][outcome].mean() - data[data[treatment] == 0]
[outcome].mean()
    print(f"Naive diff-in-means: {naive:.4f}")
    print("Causal effect estimation complete.")


def hospital_effect(df: pd.DataFrame):
    """Assess hospital-level deviations from expected survival."""
    feats = ['อายุ', 'sex', 'ถนนที่เกิดเหตุ', 'สถานะ',
             'รถผู้บาดเจ็บ', 'รถคู่กรณี', 'มาตรการ', 'การดื่มสุรา']
    sub = df[feats + ['survived', 'รหัส รพ.']].dropna()
    sub_enc = pd.get_dummies(sub, columns=feats, drop_first=True)
    X = sub_enc.drop(columns=['survived', 'รหัส รพ.'])
    y = sub_enc['survived']
    X_scaled = StandardScaler().fit_transform(X)
    log = LogisticRegression(max_iter=5000, solver='liblinear').fit(X_scaled, y)
    sub['expected'] = log.predict_proba(X_scaled)[:, 1]
    rates = sub.groupby('รหัส รพ.').agg(
        observed=('survived', 'mean'),
        expected=('expected', 'mean')
    )
    rates['difference'] = rates['observed'] - rates['expected']
    print("Hospital survival difference std:", rates['difference'].std())


def plot_hourly_survival(df: pd.DataFrame):
    """Plot survival rate by hour of day."""
    hr = (
        df[df['hour'].notnull()]
        .assign(hour=lambda x: x['hour'].astype(int))
        .groupby('hour')['survived']
        .mean()
    )
    hr.plot(title='Survival Rate by Hour', xlabel='Hour', ylabel='Survival Rate')
    plt.grid(True)
    plt.show()

def run_q1(df):
    """Run analysis for Q1: Feature Importances"""
    print("\n=== Q1: Feature Importances ===")
    feature_importance_rf(df)

def run_q2(df):
    """Run analysis for Q2: Helmet Effect"""
    print("\n=== Q2: Helmet Effect ===")
    df_mc = df[df['รถผู้บาดเจ็บ'] == 'จักรยานยนต์'].copy()
    df_mc = df_mc[df_mc['มาตรการ'].notna()]
    df_mc['helmet'] = (df_mc['มาตรการ'] == 'ใส่หมวก').astype(int)
    df_mc_enc = pd.get_dummies(df_mc, columns=['ถนนที่เกิดเหตุ','การดื่มสุรา'],
drop_first=True)
    conf = ['อายุ','sex'] + [c for c in df_mc_enc.columns if c.startswith(('ถนนที่เกิด
เหตุ_','การดื่มสุรา_'))]
    estimate_causal_effect(df_mc_enc, 'helmet', 'survived', conf)

def run_q3(df):
    """Run analysis for Q3: Seatbelt Effect"""
    print("\n=== Q3: Seatbelt Effect ===")
    car_types = ['รถเก๋ง/แท็กซี่', 'ปิคอัพ', 'รถตู้']
    df_car = df[df['รถผู้บาดเจ็บ'].isin(car_types)].copy()
    df_car = df_car[df_car['มาตรการ'].notna()]
    df_car['seatbelt'] = (df_car['มาตรการ'] == 'เข็มขัด').astype(int)
    df_car_enc = pd.get_dummies(df_car, columns=['ถนนที่เกิดเหตุ','การดื่มสุรา'],
drop_first=True)
    conf_car = ['อายุ','sex'] + [c for c in df_car_enc.columns if
c.startswith(('ถนนที่เกิดเหตุ_','การดื่มสุรา_'))]
    estimate_causal_effect(df_car_enc, 'seatbelt', 'survived', conf_car)

def run_q4(df):
    """Run analysis for Q4: Alcohol Effect"""
    print("\n=== Q4: Alcohol Effect ===")
    df_al = df[df['การดื่มสุรา'].notna()].copy()
    df_al['alcohol'] = (df_al['การดื่มสุรา'] == 'ดื่ม').astype(int)
    df_al_enc = pd.get_dummies(df_al, columns=['ถนนที่เกิดเหตุ','รถผู้บาดเจ็บ'],
drop_first=True)
    conf_al = ['อายุ','sex'] + [c for c in df_al_enc.columns if c.startswith(('ถนนที่
เกิดเหตุ_','รถผู้บาดเจ็บ_'))]
    estimate_causal_effect(df_al_enc, 'alcohol', 'survived', conf_al)

def run_q5(df):
    """Run analysis for Q5: Hospital Effect"""
    print("\n=== Q5: Hospital Effect ===")
    hospital_effect(df)

def run_q6(df):
    """Run analysis for Q6: Hourly Survival"""
    print("\n=== Q6: Hourly Survival ===")
    plot_hourly_survival(df)
```

~~~
def main():
    repo = os.path.dirname(os.getcwd())
    path = os.path.join(repo, 'final-project', 'data', 'raw.xlsx')
    df = load_data(path)
    df = clean_data(df)

    run_q1(df)
    run_q2(df)
    run_q3(df)
    run_q4(df)
    run_q5(df)
    run_q6(df)

if __name__ == '__main__':
    main()
```
~~~

Log of the first experimental run

Put everything, not remove ‘*month*’ and not binning ‘*age*’ and ‘*hour*’.

~~~
```
Duplicates: 1186
After dedup: 213764
Age 0 count: 3912
After age filter: 209852
Survival distribution:
 survived
1    206365
0      3487
Name: count, dtype: int64

=== Q1: Feature Importances ===
Top 10 factors affecting survivability:
 อายุ                       0.327995
hour                      0.231688
day_of_month              0.100451
การดื่มสุรา_ไม่ทราบ       0.043884
sex                       0.021428
ถนนที่เกิดเหตุ_ทางหลวง    0.019499
month                     0.018426
รถผู้บาดเจ็บ_ปิคอัพ       0.014926
สถานะ_ผู้ขับขี่           0.014373
รถคู่กรณี_ปิคอัพ          0.014034
dtype: float64

=== Q2: Helmet Effect ===
helmet ATE: 0.0025
Refute: Use a Placebo Treatment
Estimated effect:0.002532944948441239
New effect:-0.00017250688227805073
p value:0.88

Refute: Add a random common cause
Estimated effect:0.002532944948441239
New effect:0.002532944948441239
p value:1.0

Naive diff-in-means: 0.0076

=== Q3: Seatbelt Effect ===
seatbelt ATE: 0.0196
Refute: Use a Placebo Treatment
Estimated effect:0.019595732734418867
New effect:-0.0001993262212240316
p value:1.0

Refute: Add a random common cause
Estimated effect:0.019595732734418867
New effect:0.01959573273441887
p value:1.0
~~~

```
Naive diff-in-means: 0.0199

=== Q4: Alcohol Effect ===
alcohol ATE: 0.0064
Refute: Use a Placebo Treatment
Estimated effect:0.006412298023228655
New effect:5.520521476951851e-06
p value:0.96

Refute: Add a random common cause
Estimated effect:0.006412298023228655
New effect:0.006412298023228652
p value:1.0

Naive diff-in-means: 0.0042

=== Q5: Hospital Effect ===
Hospital survival difference std: 0.05377733820168814

=== Q6: Hourly Survival ===
(Img)
```

Log of the second experimental run

Remove ‘*month*’ and binning ‘*age*’ and ‘*hour*’.

```
Duplicates: 1186
After dedup: 213764
Age 0 count: 3912
After age filter: 209852
Survival distribution:
 survived
1    206365
0      3487
Name: count, dtype: int64

=== Q1: Feature Importances ===
Top 20 factors affecting survivability:
 day_of_month                   0.267986
hour                           0.134792
อายุ                           0.086519
การดื่มสุรา_ไม่ทราบ            0.068351
sex                            0.033993
ถนนที่เกิดเหตุ_ทางหลวง         0.030827
รถผู้บาดเจ็บ_ปิคอัพ            0.024408
ถนนที่เกิดเหตุ_ในเมือง         0.022526
รถคู่กรณี_ปิคอัพ               0.021277
มาตรการ_ไม่ทราบ                0.020664
สถานะ_ผู้ขับขี่                0.020619
รถคู่กรณี_รถเก๋ง/แท็กซี่       0.020450
มาตรการ_ไม่ใส่                 0.019593
สถานะ_ผู้โดยสาร                0.018583
การดื่มสุรา_ไม่ดื่ม            0.018096
รถผู้บาดเจ็บ_รถเก๋ง/แท็กซี่    0.017981
รถผู้บาดเจ็บ_รถจักรยาน         0.016421
รถคู่กรณี_ไม่มี/ล้มเอง         0.015624
รถคู่กรณี_ไม่ทราบ              0.015082
รถคู่กรณี_รถบรรทุก             0.014999
dtype: float64

=== Q2: Helmet Effect ===
Starting causal effect estimation...
Treatment variable: helmet
Outcome variable: survived
Number of confounders: 7
Dropping missing values...
Remaining samples after dropping NA: 167302
Creating causal model...
Identifying causal effect...
Estimating effect using PSM...
```

```
helmet ATE: 0.0113
Running refutation tests...
1. Placebo treatment refutation:
Refute: Use a Placebo Treatment
Estimated effect:0.011332799368806112
New effect:0.00017644738257761405
p value:0.8999999999999999

2. Random common cause refutation:
Refute: Add a random common cause
Estimated effect:0.011332799368806112
New effect:0.011332799368806115
p value:1.0

Computing naive difference in means...
Naive diff-in-means: 0.0075
Causal effect estimation complete.

=== Q3: Seatbelt Effect ===
Starting causal effect estimation...
Treatment variable: seatbelt
Outcome variable: survived
Number of confounders: 7
Dropping missing values...
Remaining samples after dropping NA: 19587
Creating causal model...
Identifying causal effect...
Estimating effect using PSM...
seatbelt ATE: 0.0746
Running refutation tests...
1. Placebo treatment refutation:
Refute: Use a Placebo Treatment
Estimated effect:0.07459028947771482
New effect:0.009035074283963855
p value:0.72

2. Random common cause refutation:
Refute: Add a random common cause
Estimated effect:0.07459028947771482
New effect:0.07459028947771482
p value:1.0

Computing naive difference in means...
Naive diff-in-means: 0.0191
Causal effect estimation complete.
Duplicates: 1186
After dedup: 213764
Age 0 count: 3912
After age filter: 209852
Survival distribution:
 survived
1    206365
0      3487
Name: count, dtype: int64

=== Q4: Alcohol Effect ===
```

~~~
Starting causal effect estimation...
Treatment variable: alcohol
Outcome variable: survived
Number of confounders: 17
Dropping missing values...
Remaining samples after dropping NA: 209851
Creating causal model...
Identifying causal effect...
Estimating effect using PSM...
alcohol ATE: 0.0060
Running refutation tests...
1. Placebo treatment refutation:
Refute: Use a Placebo Treatment
Estimated effect:0.006018556023083045
New effect:0.00039408913943702906
p value:0.96

2. Random common cause refutation:
Refute: Add a random common cause
Estimated effect:0.006018556023083045
New effect:0.006018556023083047
p value:1.0

Computing naive difference in means...
Naive diff-in-means: 0.0038
Causal effect estimation complete.

=== Q5: Hospital Effect ===
Hospital survival difference std: 0.05340669455562482

=== Q6: Hourly Survival ===
(Img)
```
~~~

GitHub Repository

The full implementation of this project, including data preprocessing, causal analysis code, result logs, and visualizations, is available on GitHub `repository: pupipatsk/Causal-Inference-and-Discovery`. https://github.com/pupipatsk/Causal-Inference-and-Discovery.git