

基本模型 (OLS)

Temperature (溫度): 係數為-0.0018，但不顯著 (P 值為 0.255)

Rainfall (降雨量): 係數為-0.0021，且顯著 (P 值為 0.000)

Play_nWeek (上映週數): 係數為-0.0120，顯著 (P 值為 0.000)

Weekend (週末): 係數為 0.3610，顯著 (P 值為 0.000)

Director_Award (導演獲獎): 係數為-0.0417，顯著 (P 值為 0.000)，表明導演獲獎有負面影響，這可能反映了某些導演獲獎影片的市場吸引力較低。

Producer_Famous (知名製片人): 係數為-0.0926，顯著 (P 值為 0.000)，表明知名製片人對有負面影響，可能是因為知名製片人參與的電影並不一定能夠吸引更多觀眾。

Budget (預算): 係數為 1.508e-08，顯著 (P 值為 0.000)

其他變數: 包括 Actor_Awardall、IMDb_rating、Oscar_Nom、Cannes_Nom 等，這些變數對 Log_Audience 也有顯著影響。

DID

處理組 (Treatment Group)

處理組是那些受到政策 (Policy) 或疫情 (Pandemic) 影響的觀測值：

```
data['treatment'] = ((data['Policy'] == 1) | (data['Pandemic'] == 1)).astype(int)
```

treatment 變數為 1 的觀測值屬於處理組，即在政策或疫情期間的觀測值。

對照組 (Control Group)

對照組是那些沒有受到政策或疫情影響的觀測值：

treatment 變數為 0 的觀測值屬於對照組，即不在政策或疫情期間的觀測值。

時間維度的分組

除了處理組和對照組，還有時間上的分組，即在 2020 年及以後的觀測值與之前

的觀測值：

```
data['post'] = (data['PlayYear'] >= 2020).astype(int)
```

post 變數為 1 的觀測值屬於 "後期" (即 2020 年及以後)，為 0 的觀測值屬

於 "前期" (即 2020 年以前)。

結合處理組和時間分組

DID 模型：

```
did_formula = "
```

```
Log_Audience ~ treatment * post + Tempture + Rainfall + Weekend +  
Director_Award + Producer_Famous + Budget + Actor_Awardall + IMDb_rating +  
Oscar_Nom + Cannes_Nom + GHA_Nom + GHA_AwardTotal + TFF_Nom +  
TFF_AwardTotal + Insubsidy + CSP + SD + SpecialCondition + Drama_Comedy +  
Action_Comedy + Typhoon + Earthquake + Policy + Festival + Promotion +  
Pandemic + Temp_Spring + Temp_Summer + Temp_Winter + Rain_Spring +  
Rain_Summer + Rain_Winter  
"
```

```
did_model = smf.ols(did_formula, data=data).fit()
```

使用 treatment * post 交互項來捕捉處理組在不同時間段的效果。：

$treatment = 1$ 且 $post = 1$ ：代表處理組在 2020 年及以後。

$treatment = 1$ 且 $post = 0$ ：代表處理組在 2020 年以前。

$treatment = 0$ 且 $post = 1$ ：代表對照組在 2020 年及以後。

$treatment = 0$ 且 $post = 0$ ：代表對照組在 2020 年以前。

處理組： $treatment = 1$ （受到政策或疫情影響的觀測值）。

對照組： $treatment = 0$ （未受到政策或疫情影響的觀測值）。

後期： $post = 1$ （2020 年及以後的觀測值）。

前期： $post = 0$ （2020 年以前的觀測值）。

DID

$treatment$ (處理): 係數為 0.8117，顯著 (P 值為 0.000)，表明受到政策或疫情影響的觀測值對 $Log_Audience$ 有顯著正向影響。

$post$ (後期): 係數為 -0.1194，顯著 (P 值為 0.007)，表明 2020 年及以後的時間段對 $Log_Audience$ 有顯著負向影響。

交互項 $treatment * post$ ：係數為 -0.1194，顯著 (P 值為 0.007)，表明處理組在後期相比於前期有顯著的負面影響，這意味著政策或疫情在 2020 年及以後對觀眾數量的影響是負面的。

Tempature (溫度): 係數為 0.0119，顯著 (P 值為 0.000)，表明溫度對

Log_Audience 有顯著的正向影響，這與基本模型中的結果不同。

Weekend (週末): 係數為 0.5937，顯著 (P 值為 0.000)，表明週末上映對

Log_Audience 有顯著的正向影響。

其他變數: 多數變數如 IMDb_rating、Oscar_Nom 等在模型中也是顯著的，並

且對 Log_Audience 有預期的影響。

週末和較高的 IMDb 評分、奧斯卡提名等對於提升電影觀眾數量有顯著正向影

響。降雨量和某些獲獎導演或知名製片人的電影，可能因為類型或其他因素對

觀眾吸引力有限，反而有負面影響。在 DID 模型中，處理組在後期相比前期的

影響顯著負向，這意味著政策或疫情在 2020 年及以後對觀眾數量有不利影響。

這些結果提供了對於不同因素如何影響電影觀眾數量，有助於在制定電影市場

策略時考慮這些變數的影響。

分量迴歸：處理不同分量的票房

在不同的分位數下探索不同因素對電影觀眾數量 (Log_Audience) 的影響：

創建所需的變數：

```

data['CSP'] = ((data['PlayYear'] == 2020) & (data['PlayMonth'].between(3,
6))).astype(int) ( 梅花座政策 )

data['SD'] = ((data['PlayYear'] == 2020) & (data['PlayMonth'].between(4,
6))).astype(int) ( 社交距離政策 )

data['SpecialCondition'] = ((data['PlayYear'] == 2020) &
(data['PlayMonth'].between(1, 4))).astype(int) ( 特殊情況 )

data['Drama_Comedy'] = ((data['Class_Darma'] == 1) & (data['Class_Comedy'] ==
1)).astype(int) ( 同時為劇情和喜劇 )

data['Action_Comedy'] = ((data['Class_Action'] == 1) & (data['Class_Comedy'] ==
1)).astype(int) ( 同時為動作和喜劇 )

```

定義特定日期範圍的事件：

`typhoon_dates` = [('2017-07-01', '2017-07-05'), ...] 颱風

`earthquake_dates` = [('2018-02-06', '2018-02-08'), ...] 地震

`policy_dates` = [('2017-01-01', '2017-12-31'), ...] 政策通常為影響一整年

`festival_dates` = [('2017-01-01', '2017-02-28'), ...] 國訂假日

`promotion_dates` = [('2017-07-01', '2017-07-31'), ...] 暑假電影節促銷、暑假強檔

`pandemic_dates` = [('2020-01-21', '2020-12-31'), ...] 疫情發生

創建 Season 變數並轉換為虛擬變數：

```

data['Season'] = data['PlayMonth'].apply(lambda x: 'Spring' if x in [3, 4, 5] else
('Summer' if x in [6, 7, 8] else ('Autumn' if x in [9, 10, 11] else 'Winter')))
data = pd.get_dummies(data, columns=['Season'], drop_first=True)

```

新增季節性效果，創建季節和溫度、降雨交互項：

```

data['Temp_Spring'] = data['Temperture'] * data['Season_Spring']

```

```
data['Temp_Summer'] = data['Tempature'] * data['Season_Summer']
data['Temp_Winter'] = data['Tempature'] * data['Season_Winter']
data['Rain_Spring'] = data['Rainfall'] * data['Season_Spring']
data['Rain_Summer'] = data['Rainfall'] * data['Season_Summer']
data['Rain_Winter'] = data['Rainfall'] * data['Season_Winter']
```

創建 Log_Audience 變數：將票數取對數（為何不選票房因為每間電影院票價不一樣），因為票數為右偏將其取對數後較符合常態

```
data['Log_Audience'] = data['Audience'].apply(lambda x: np.log(x + 1))
```

創建交互項：看有無缺失值和該變數的交互作用（這邊沒效果可以忽略）

```
data['Subsidy_Yes:lnbudget'] = data['Subsidy_Yes'] * data['lnbudget']
data['Subsidy_Yes:lnsubsidy'] = data['Subsidy_Yes'] * data['lnsubsidy']
data['No_Budget:lnbudget'] = data['No_Budget'] * data['lnbudget']
data['No_IMDb:IMDb_rating'] = data['No_IMDb'] * data['IMDb_rating']
data['No_GHA:GHA_Nom'] = data['No_GHA'] * data['GHA_Nom']
data['No_GHA:GHA_AwardTotal'] = data['No_GHA'] * data['GHA_AwardTotal']
data['No_TFF:TFF_Nom'] = data['No_TFF'] * data['TFF_Nom']
data['No_TFF:TFF_AwardTotal'] = data['No_TFF'] * data['TFF_AwardTotal']
```

建議：

票價 (Ticket_Price)：

票價的變化可能會影響觀眾數量，特別是在促銷期間或節假期間。

競爭影片 (Competing_Movies)：

同一時間段內上映的其他影片數量和它們的影響力。

社交媒體評價 (Social_Media_Rating)：（電子口碑 EWOM）

社交媒體上的評價和討論度，如 Twitter 和 Facebook 上的評論數量和情感分析結果。

經濟狀況 (Economic_Conditions) :

當前經濟狀況，如失業率和消費者信心指數等。