員工會跳槽嗎?

資工四

4110056029 劉菡侖

4110056030 鄭詠謙

Agenda

1 資料預處理

2 模型選擇

3 系統架設

Contents:

研究目的

希望利用公司內部數據建立一個有效的分析模型
預測員工的**離職傾向**及其**可能的影響因素**

分析多維度數據

找出潛在的離職員工

採取針對性留任策略
降低人力流失成本

資料集介紹

資料來源：kaggle

Categorical (String)	Integer	Integer (1-10 scale)	Integer (0 or 1)
Field of Study	Age	Job Satisfaction	Career Change Interest
Current Occupation	Years of Experience	Work-Life Balance	Mentorship Available
Gender	Salary	Job Security	Certifications
Education Level	Career Change Events	Professional Networks	Freelancing Experience
Family Influence	Job Opportunities	Technology Adoption	Geographic Mobility
Industry Growth Rate		Skills Gap	Likely to Change Occupation

Number of Rows: 38,444
Number of Features: 22

資料分佈

觀察資料分布

是否存在

- 資料不平衡
- 缺值
- 離群值

進行預處理

預處理

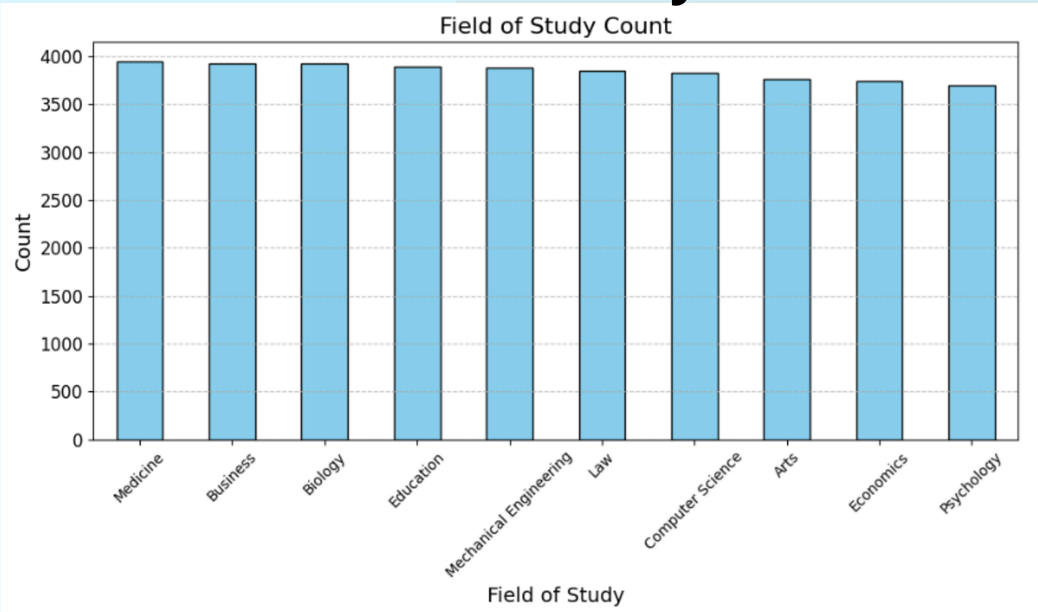
- 補缺失值
- 刪除離群值

文字類別型資料

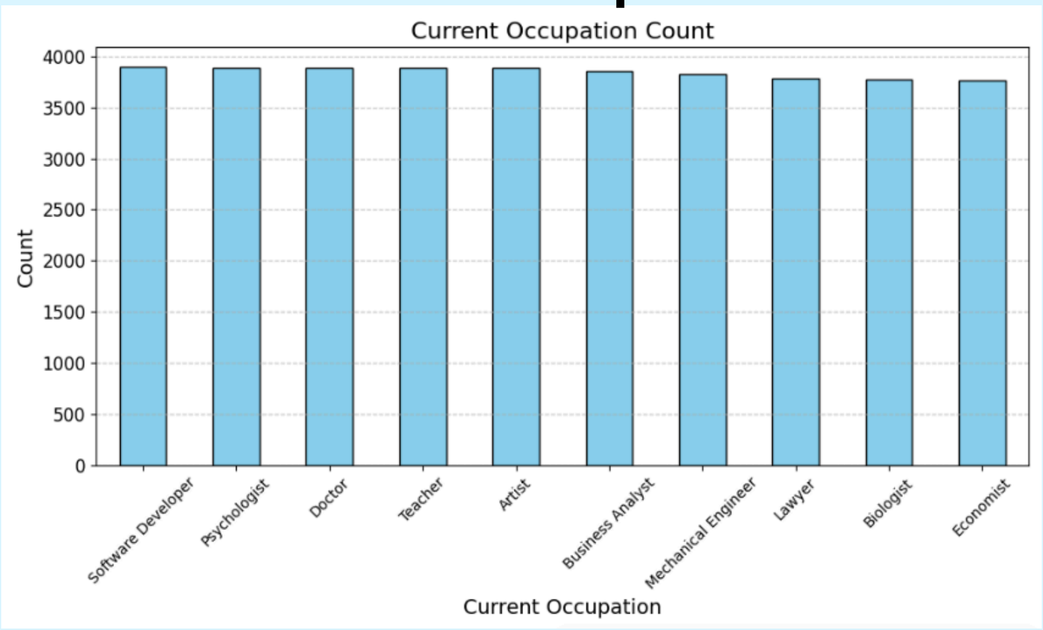
- One-Hot Encoding

資料分佈

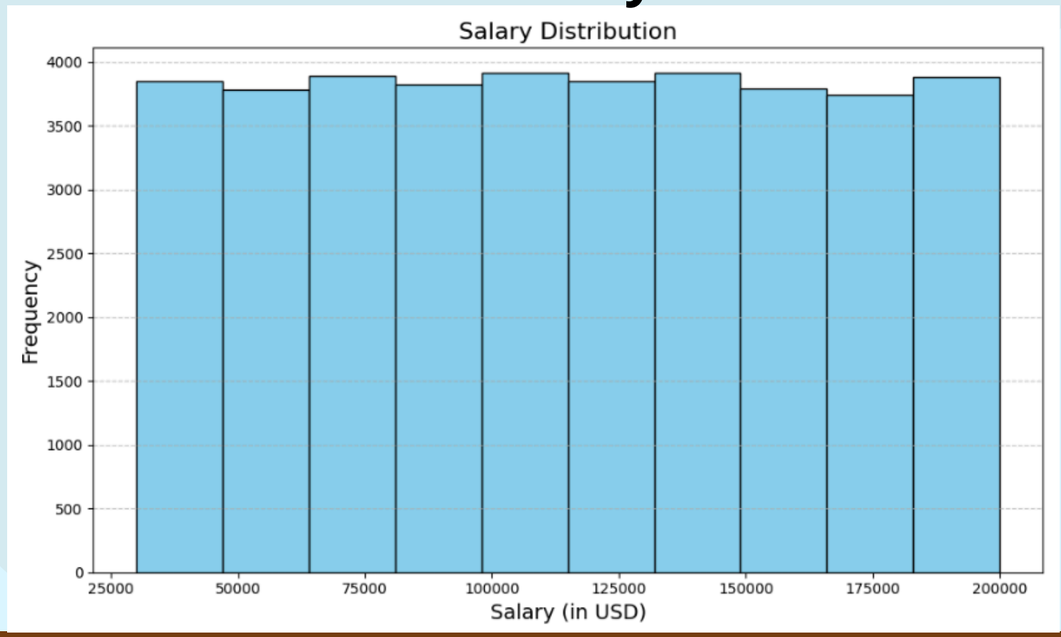
Field of Study



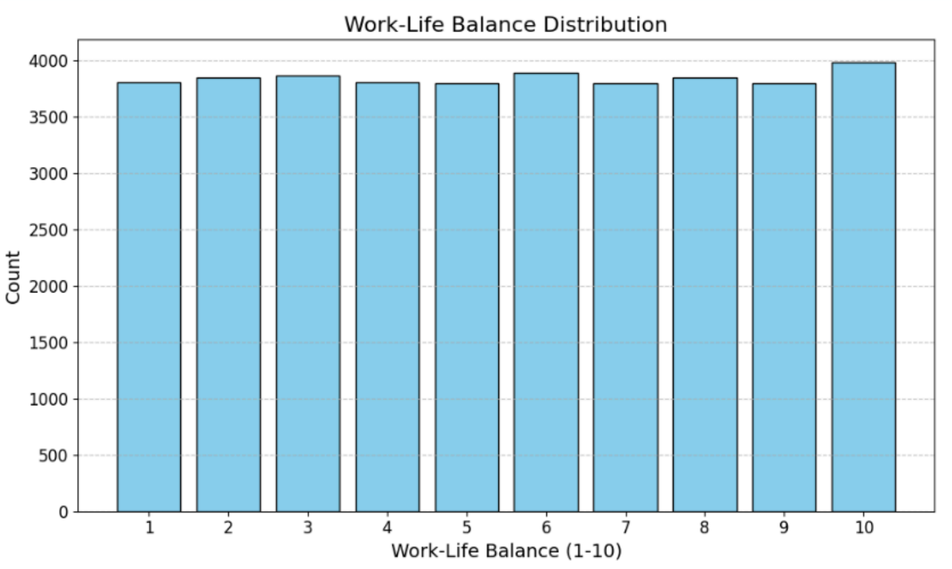
Current Occupation



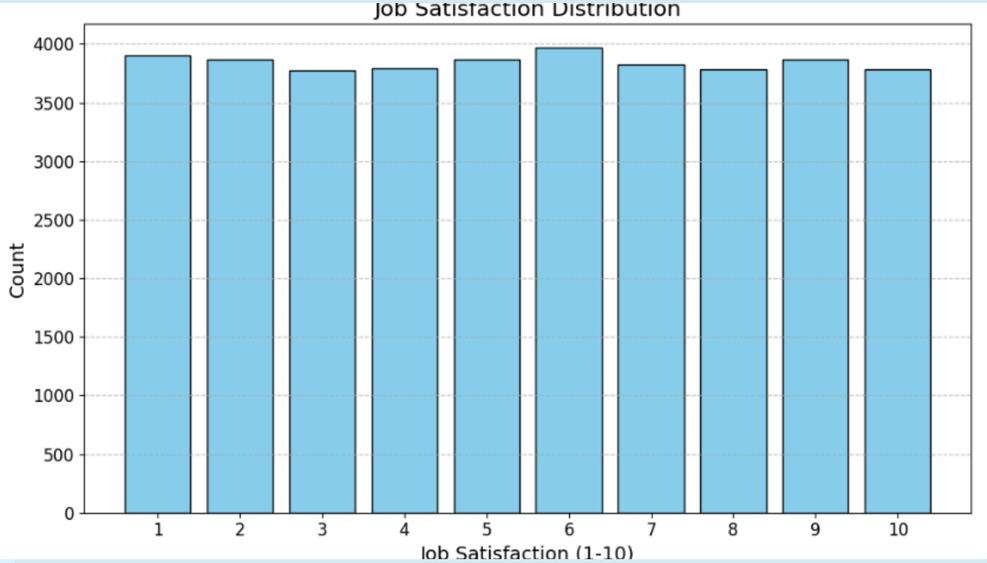
Salary



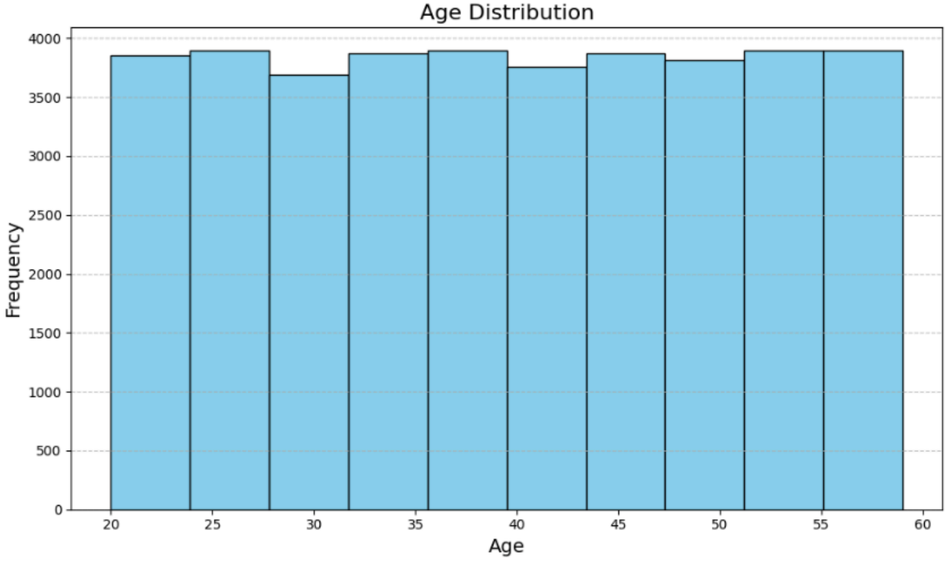
Work-Life Balance



Job Satisfaction

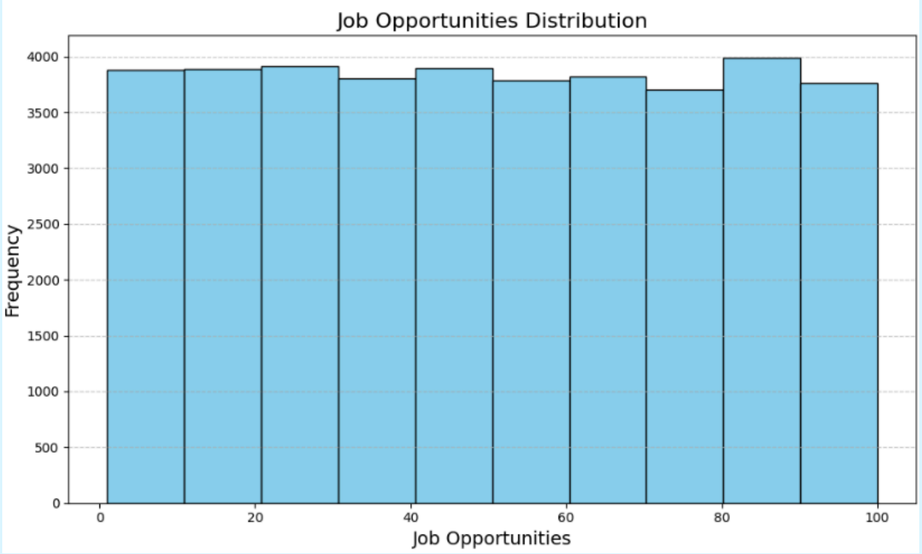


Age

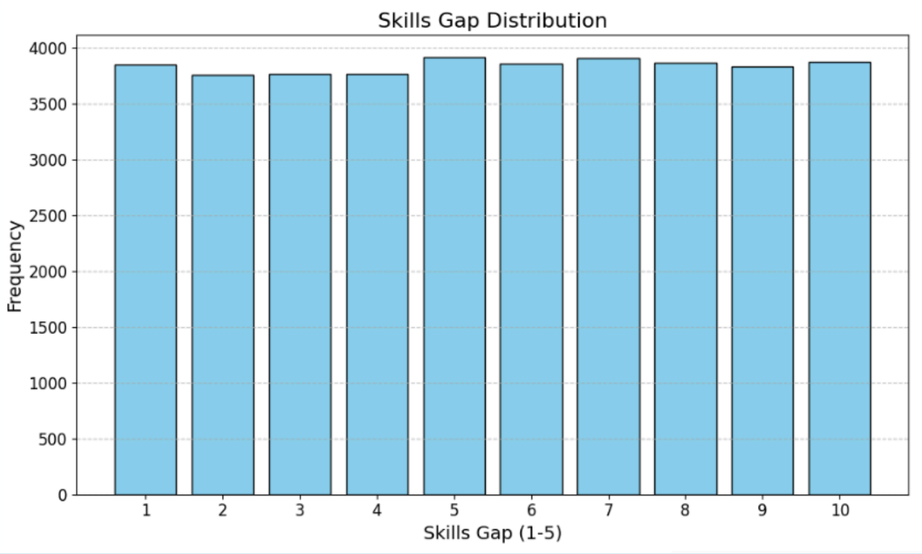


Years of Experience

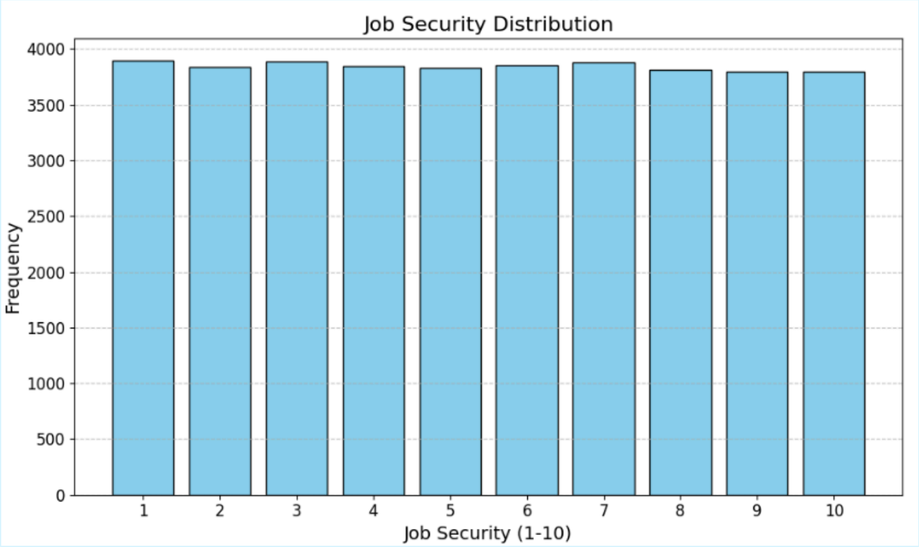
Job Opportunities



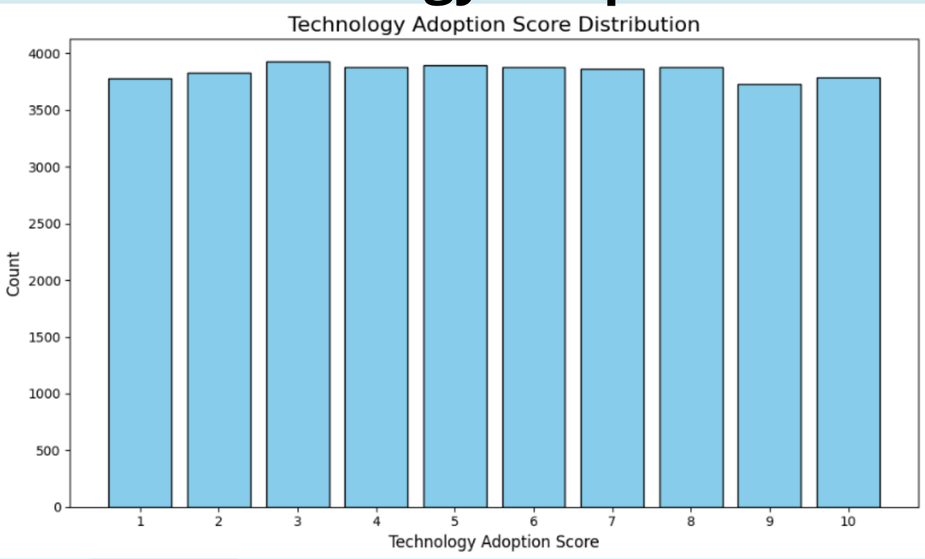
Skills Gap



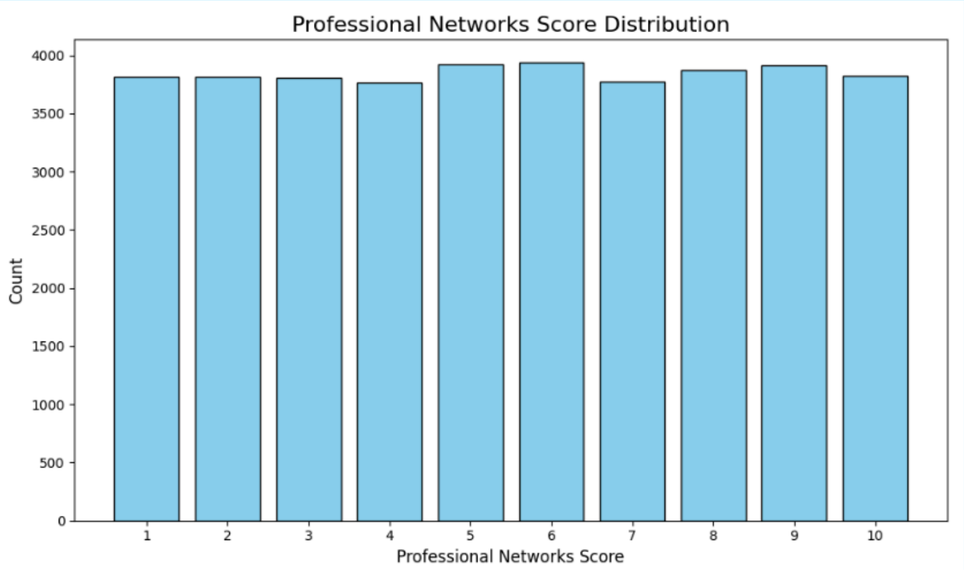
Job Security



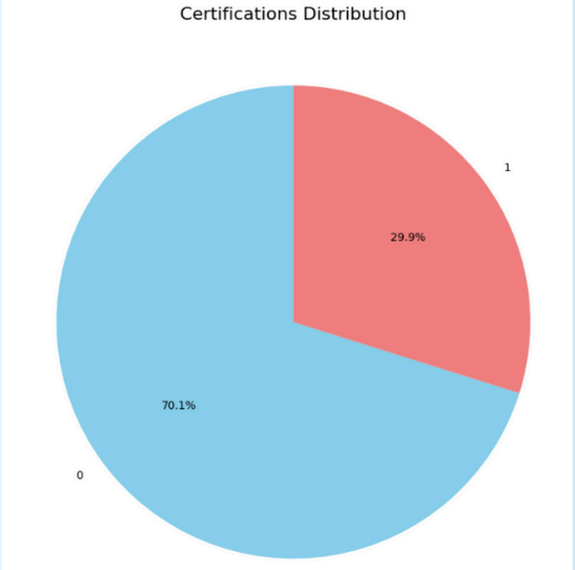
Technology Adoption



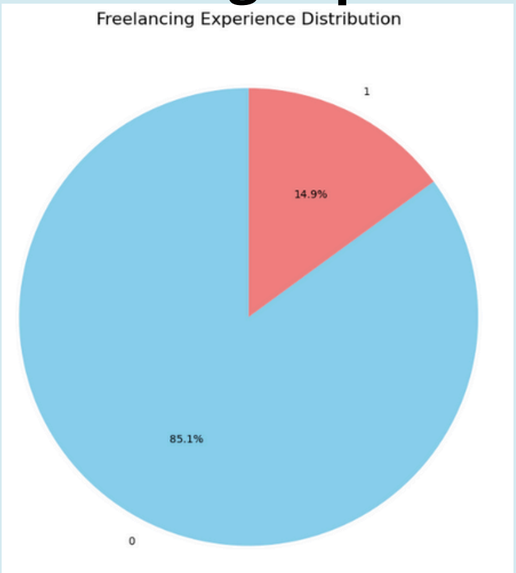
Professional Networks



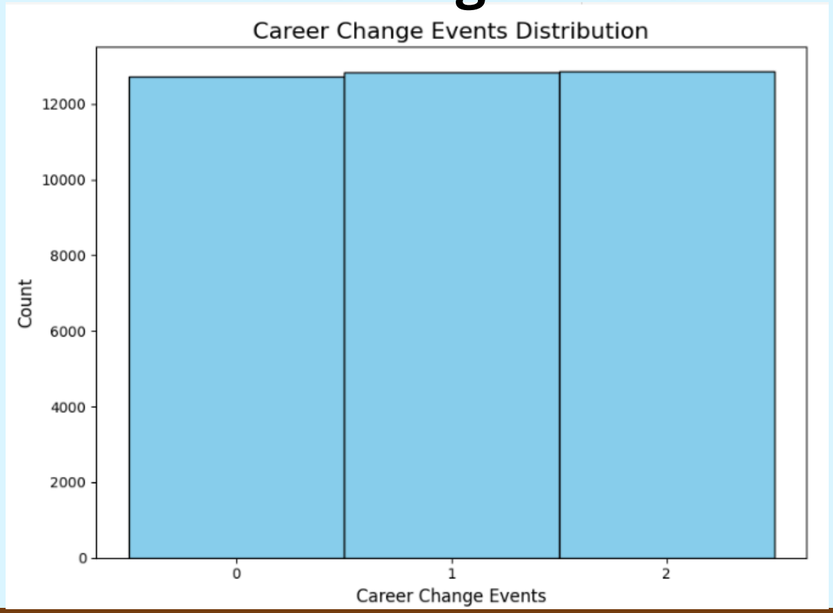
Certifications



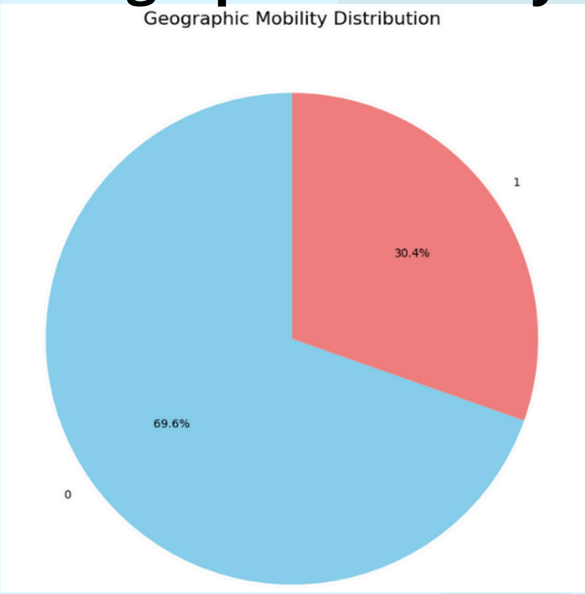
Freelancing Experience



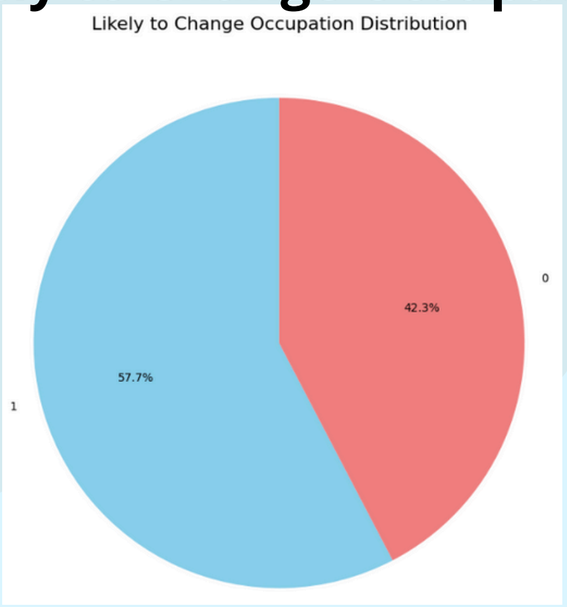
Career Change Events



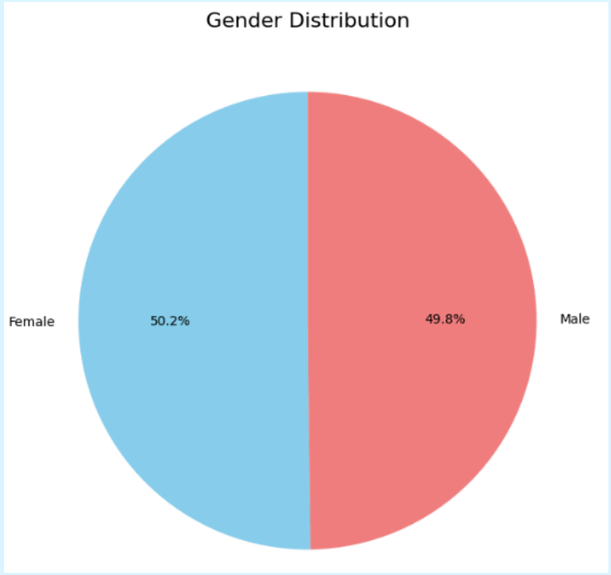
Geographic Mobility



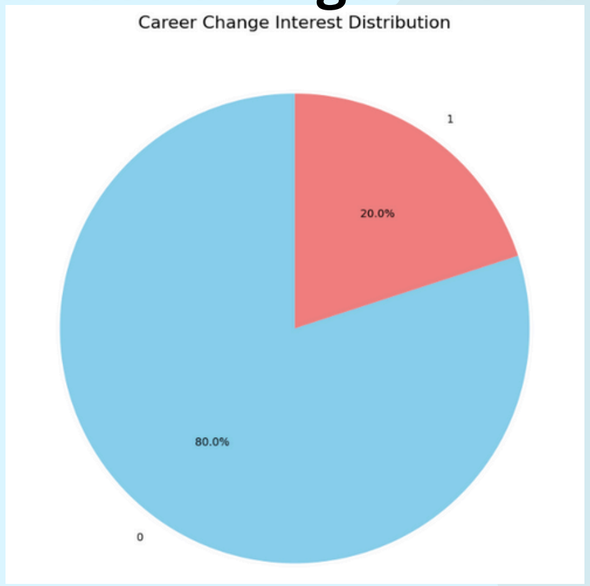
Likely to Change Occupation



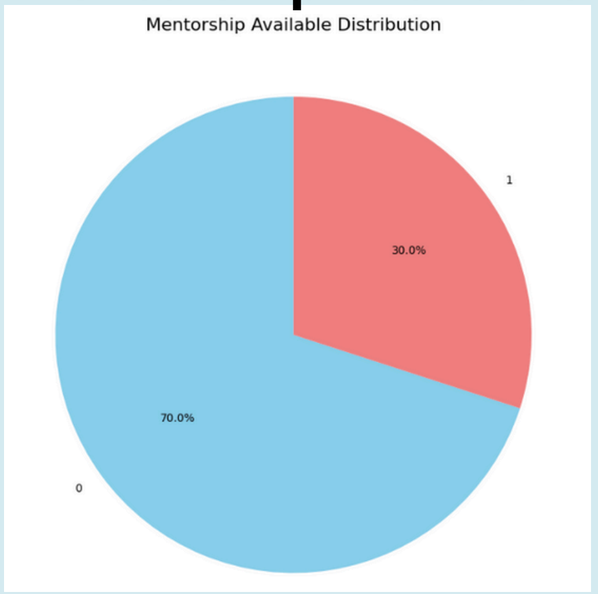
Gender



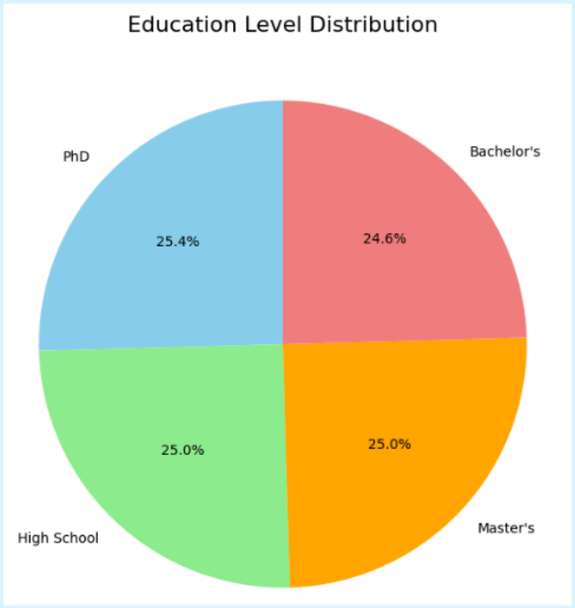
Career Change Interest



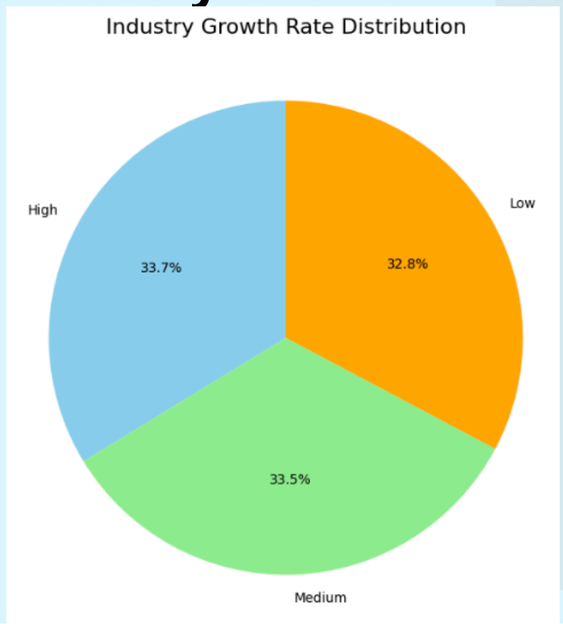
Mentorship Available



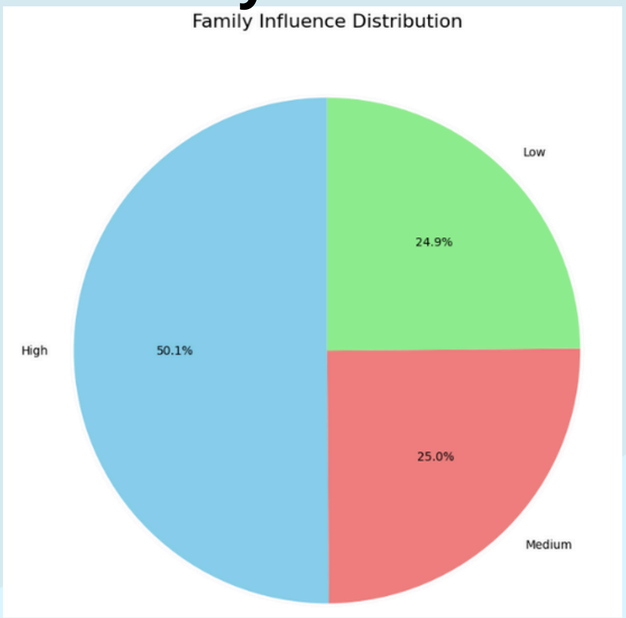
Education Level



Industry Growth Rate

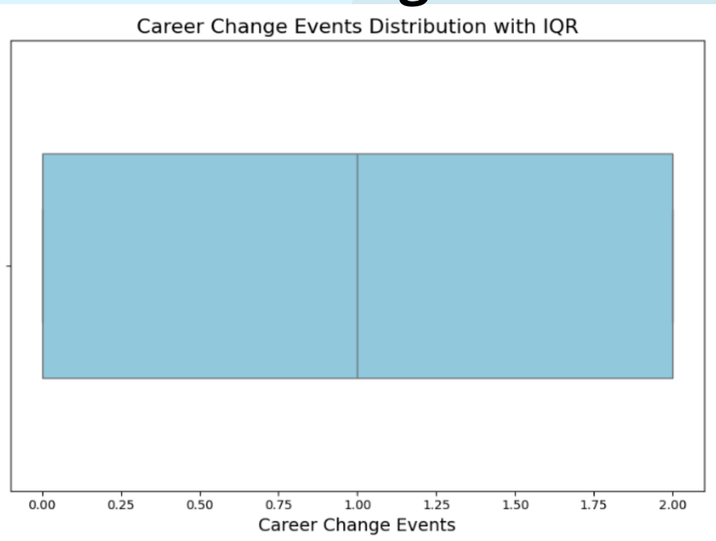


Family Influence

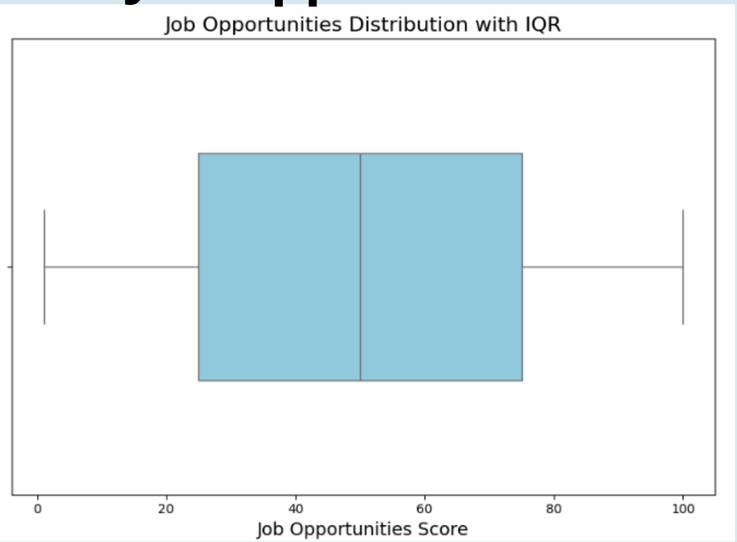


離群值檢查

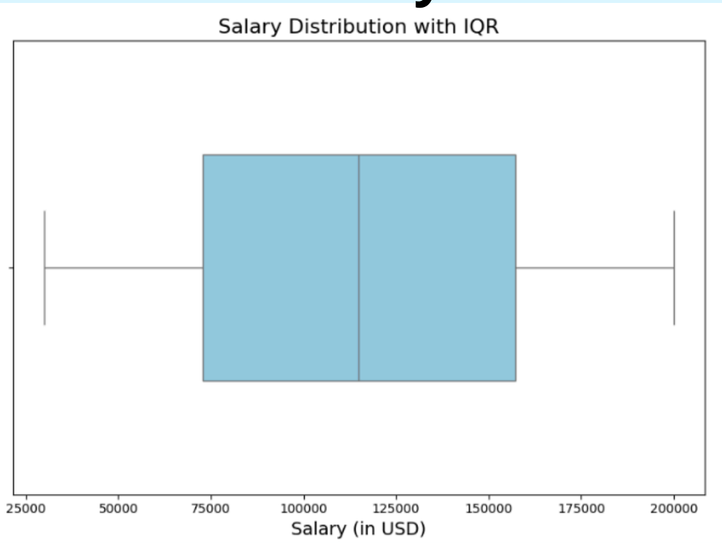
Career Change Events



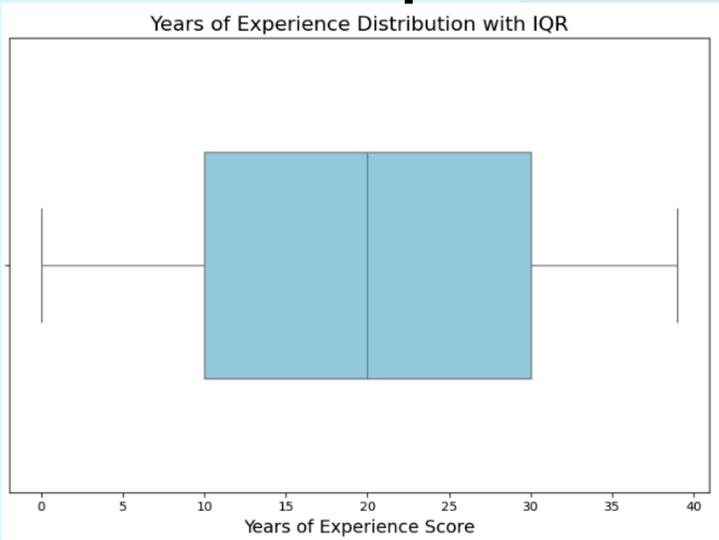
Job Opportunities



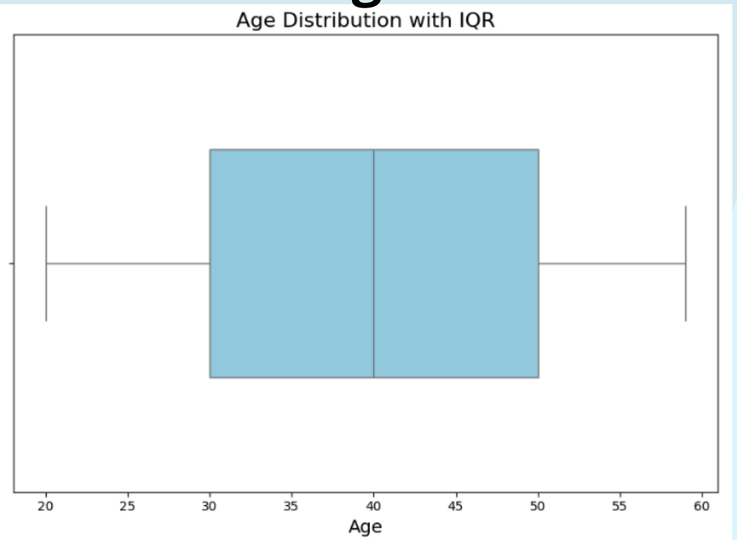
Salary



Years of Experience



Age



劃分數據

訓練集：測試集 = 7：3

```
X_train_s, X_test_s, y_train, y_test = train_test_split(X_selected, y, test_size=0.3, random_state=42)
```

X_train.shape, y_train.shape, X_test.shape, y_test.shape

```
((26910, 22), (26910,), (11534, 22), (11534,))
```

模型訓練

All model:

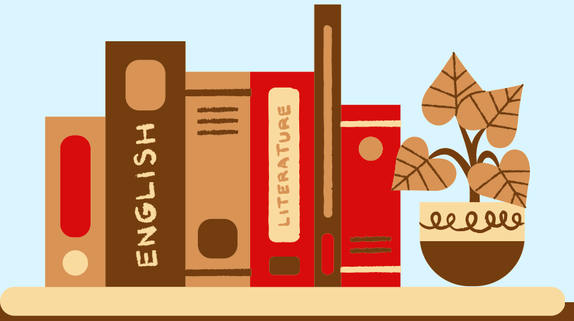
- **Random Forest**
- **Gradient Boosting**
- **XGBoost**
- **Logistic Vector Machine**
- **Naive Bayes**
- **K-Nearest Neighbors**
- **AdaBoost**
- **Bagging**
- **Decision Tree**

```
models = {
        'Random Forest': RandomForestClassifier(random_state=42),
        'Gradient Boosting': GradientBoostingClassifier(random_state=42),
        'XGBoost': XGBClassifier(random_state=42),
        'Logistic Regression': LogisticRegression(random_state=42),
        'Support Vector Machine': SVC(probability=True, random_state=42),
        'Naive Bayes': GaussianNB(),
        'K-Nearest Neighbors': KNeighborsClassifier(),
        'AdaBoost': AdaBoostClassifier(random_state=42),
        'Bagging': BaggingClassifier(random_state=42),
        'Decision Tree': DecisionTreeClassifier(random_state=42)
}
```

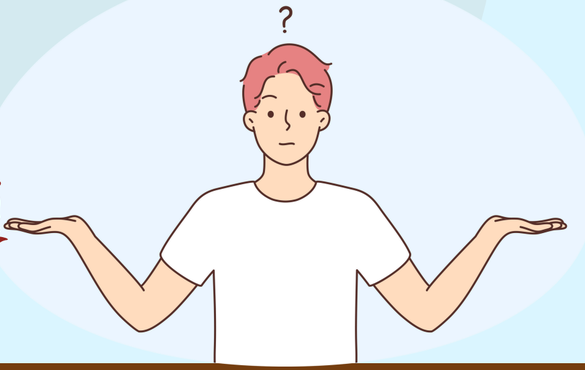
發現問題

第一次訓練結果

	Accuracy	Precision	Recall	F1 Score	ROC AUC
Random Forest	1.000000	1.000000	1.000000	1.000000	1.000000
Gradient Boosting	1.000000	1.000000	1.000000	1.000000	1.000000
XGBoost	0.999870	0.999777	1.000000	0.999888	0.999982
Logistic Regression	0.764339	0.804016	0.787134	0.795485	0.845552
Support Vector Machine	0.582260	0.582260	1.000000	0.735986	0.525788
Naive Bayes	0.800624	0.833182	0.822202	0.827656	0.899978
K-Nearest Neighbors	0.561972	0.623745	0.624302	0.624023	0.596107
AdaBoost	1.000000	1.000000	1.000000	1.000000	1.000000
Bagging	1.000000	1.000000	1.000000	1.000000	1.000000
Decision Tree	1.000000	1.000000	1.000000	1.000000	1.000000



→發現訓練結果可能存在「過擬和」問題



進行優化

我們使用K-Fold Cross-Validation來評估模型，並且降低模型複雜度（減少樹的數量（n_estimators)，增加葉子節點的最小樣本數（min_samples_leaf）...等），但結果依舊

故我們決定改變方向，往可能存在「特徵工程」問題來進行改善，特徵可能包含過多資訊，甚至有「目標變量洩露」的情況

特徵選擇

使用部分欄位來進行預測

降低模型複雜度：

- 特徵數量少可以降低計算成本，減少過擬合風險
- 模型更簡單，結果更容易解釋，適合公司人力資源業務應用

特徵選擇後的效能：

- 如果相關性分析、業務邏輯和模型測試已證實這些特徵的有效性，使用部分欄位即可
- 不需要因小特徵貢獻而額外增加噪音

特徵選擇

1 我們發現有一個特徵與目標變量高度相關"Career Change Interest"，
故將此特徵移除

2 檢查特徵重要性，去除冗餘或無用的特徵：
最後留下"Job Satisfaction"、"Salary"，剩餘特徵相關性都非常低

3 從公司的角度來看，讓公司預測員工是否會離職的目標，
應該不僅考慮數據中的相關性高低，還要考慮業務背景和特徵的實際解釋力，
所以我們又選擇3個特徵：
"Work-Life Balance"、"Job Security"、"Industry Growth Rate"

最終選擇的5個特徵：
"Job Satisfaction"、"Salary"、"Work-Life Balance"、
"Job Security"、"Industry Growth Rate"

模型重新訓練

All model:

- **Random Forest**
- **Gradient Boosting**
- **XGBoost**
- **Logistic Vector Machine**
- **Naive Bayes**
- **K-Nearest Neighbors**
- **AdaBoost**
- **Bagging**
- **Decision Tree**

```
models = {
        'Random Forest': RandomForestClassifier(random_state=42),
        'Gradient Boosting': GradientBoostingClassifier(random_state=42),
        'XGBoost': XGBClassifier(random_state=42),
        'Logistic Regression': LogisticRegression(random_state=42),
        'Support Vector Machine': SVC(probability=True, random_state=42),
        'Naive Bayes': GaussianNB(),
        'K-Nearest Neighbors': KNeighborsClassifier(),
        'AdaBoost': AdaBoostClassifier(random_state=42),
        'Bagging': BaggingClassifier(random_state=42),
        'Decision Tree': DecisionTreeClassifier(random_state=42)
}
```

模型選擇與評估

著重的點 >>>> "要盡量找出可能離職的人" (class 1)

挑選模型標準 >>>> "recall" 為主，同時兼顧 "precision" & "F1 score"

	Precision	Recall	F1 Score
Random Forest	0.930233	0.830914	0.877773
Gradient Boosting	0.999453	0.815725	0.898290
XGBoost	0.989195	0.817958	0.895464
Logistic Regression	0.807991	0.813044	0.810510
Support Vector Machine	0.971007	0.822872	0.890823
Naive Bayes	0.826613	0.824213	0.825411
K-Nearest Neighbors	0.919703	0.828903	0.871945
AdaBoost	0.999453	0.815725	0.898290
Bagging	0.953314	0.825553	0.884846
Decision Tree	0.841144	0.853920	0.847484

將多個model做precision recall F1-score分析

以"recall"值來說: "Decision Tree"表現最好，第二則是"Random Forest"

"Decision Tree"的"precision"、"F1 Score"太低 → 最終選擇"Random Forest"模型

模型選擇與評估

最終模型結果

Random Forest Classification Report:

	precision	recall	f1-score	support
0	0.79	0.91	0.85	3212
1	0.93	0.83	0.88	4477
accuracy			0.87	7689
macro avg	0.86	0.87	0.86	7689
weighted avg	0.87	0.87	0.87	7689

~ Demo Time ~

模型部署與測試 - GUI / 網頁

Please refer to the attached video file