

元智大學資訊管理系
決策支援系統期末專題報告

研究主題：
薪資預測與職涯規劃

指導老師：詹前隆老師

1091603 池昀憶

1091606 曾暄雅

1091613 林芸晨

1091728 蔡芊葳

1091752 黃湘樺

目錄

一、計畫概述	1
二、簡介	2
(一)、研究背景與動機	2
(二)、研究目的	3
三、文獻探討	4
四、研究方法	5
(一)、使用軟體	5
(二)、隨機森林迴歸	5
(三)、模型評估指標	7
五、系統建構	8
(一)、資料取得和說明	8
(二)、資料前處理	8
(三)、資料集內容	10
(四)、資料庫建構	11
(五)、隨機森林模型建構	13
(六)、介面程式開發	14
(七)、系統說明	15
六、結論與建議	18
(一)、預期效益	18
(二)、結果展示	18
(三)、問題與改進方向	19
七、人員分配	21
八、參考資料	21

一、計畫概述

臨近畢業，應屆畢業生薪資話題居高不下，對於新鮮人來說，要找到滿意的工作不容易，人力銀行發布的「九年級生待業痛苦與上工開心指數」調查顯示有 80.2% 的新鮮人對目前的收入感到不滿意，而對於擁有正職的新鮮人中，有 70.1% 在工作內容方面缺乏成就感，綜合評估薪資滿意度和工作成就感後，新鮮人自評的「上工開心指數」僅為 41.2 分。顯示出整體而言，新鮮人在工作上的滿意度相對較低，其中有 15.1% 的新鮮人給予自評為「零分」，這反映了一部分新鮮人對於目前工作狀態的極度不滿。

本研究根據 Dcard 各公司現職就職者表單回饋內容分析職場中影響薪資的各種因素，進行相關單位校正、處理缺失值，篩選樣本數 368 筆並選定特徵變量與目標變量。研究目標以科技業為主，蒐集科技業不同職種的工作情況，設計一個基於隨機森林的預測模型，該模型利用求職者選擇的特定公司名稱和職務輸入的預期工作條件，如每月平均工時、加班頻率、上班心情和工作量，來預測薪資水平，結合 Python 程式的 Tkinter 套件介面，回傳深入該職業後可獲得的預期薪資，以成為使用者在考量工作職位時的薪資參考依據。旨在幫助求職者進行選擇職種或是公司時，能夠挑選最適合自己的職涯發展方向，減少在工作上不滿意的情況，協助求職者在工作量、薪資和年資上做平衡，選擇到適合自己的工作。

二、簡介

(一)、研究背景與動機

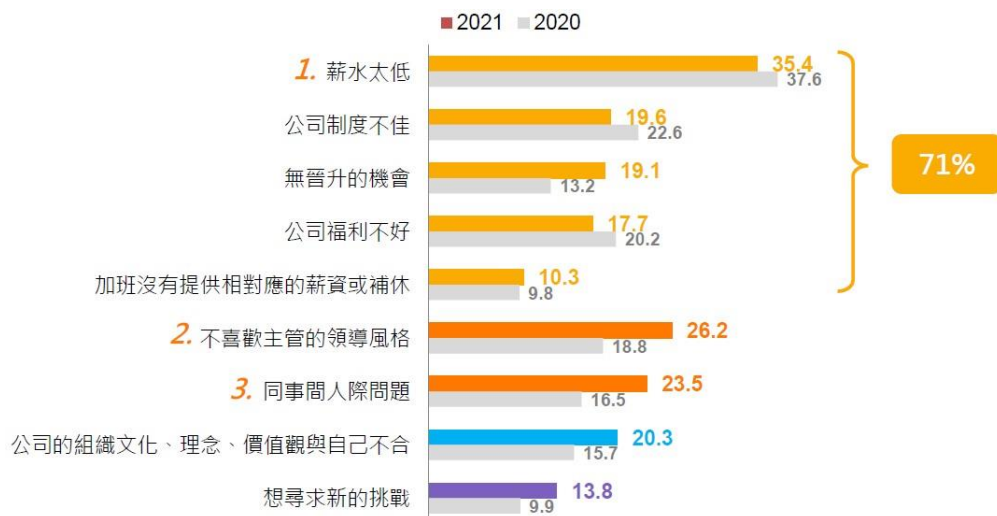
即將畢業的我們要踏入職場，對於工作上的一些問題充滿好奇，因此也希望透過此次研究，利用資料蒐集以及系統的製作，更加深入了解職場中可能面臨到大大小小的事。

當談到就業，人們關心的問題百百種，不管是應屆畢業生或是已工作的人都會因為各種因素懷疑是否該換新公司更或是轉換新跑道。

對於新鮮人和想轉換公司的求職者薪資是一大考慮因素，而根據圖一顯示人們在考慮轉換公司時最大原因則是「薪資太低」。因此，了解薪資增長的情況成為留在同一行業的重要考量之一。

然而，在職場中，薪資話題常被視為一個忌諱，而這種忌諱卻是影響職業發展的重要因素之一。許多人對於討論薪資感到不自在，但薪資卻是職業發展中極為關鍵的考量。

因此，本研究的研究動機在於填補這種在職場中存在的「薪資話題忌諱」的空白，致力於打破這種不確定性，提供用戶薪資預測，讓他們能夠透過了解工作各項數值與薪資漲幅之間的關聯，可以更好地規劃自己的職業生涯，並有更明確的目標與期許。



圖一、2020 與 2021 轉換公司比例示意圖

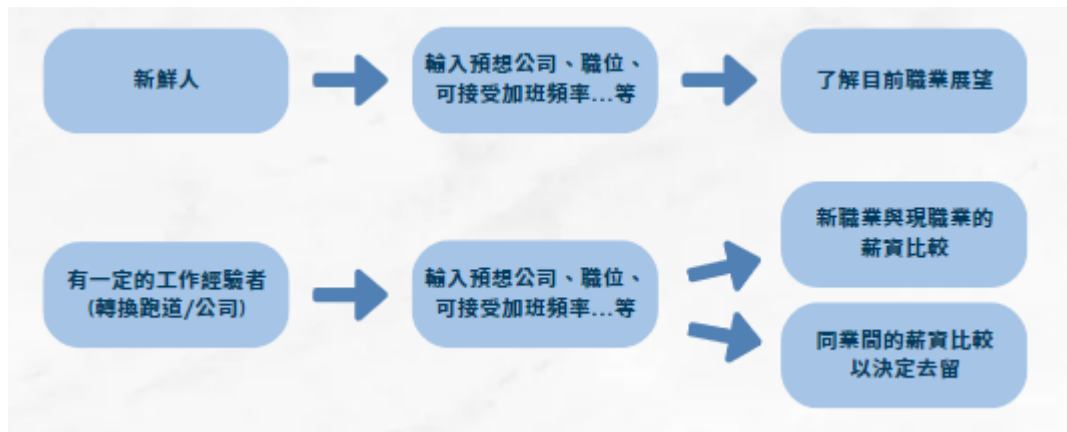
(二)、研究目的

本研究旨在建立一個以隨機森林迴歸為基礎的薪資預測模型，如圖二以滿足兩大主要對象的需求：新鮮人和有一定工作經驗者。

新鮮人在求職過程中，可以透過選擇特定公司名稱和職務以及輸入預想的每月平均工時、加班頻率、上班心情和工作量等來預測薪資，將此視為未來職涯的參考指標，也可應用於面試時的期望薪資設定。

對於有一定工作經驗者可透過本系統填入預想的公司、職務，了解不同公司、職位、年資在薪資中的成長幅度，根據同業資料薪酬的比較，使用者能夠更準確地評估自己目前的薪資水平是否合理，參考公司的薪資調整確保其付出與所得相符合，以此預測結果作為是否轉換跑道或跳槽的參考依據，讓使用者更有把握的做出職業發展的決策。

研究將透過資料蒐集、分析及建模，提供使用者對於未來職業薪資的預測，讓使用者在選擇職業時也加入行業目前薪資的發展趨勢做為參考，以填補薪資話題在職場中的不確定性，幫助評估一份工作的吸引力，衡量工作量與薪資是否符合自身需求，為使用者提供更具信心與可行性的職涯規劃工具。



圖二、研究對象和目的示意圖

三、文獻探討

台灣薪資預測文獻

廖宜川（2020）的研究使用爬蟲技術抓取台灣資訊軟體業人員的人力銀行資訊，並建構了一個隨機森林薪資預測模型，旨在幫助求職者了解影響薪資的因素，同時提供企業方降低雙方間摩擦的參考依據。而本研究將使用 Dcard 表單調查之開放資料集，參考該論文預測模型與變數，針對科技業中各種職位之薪資進行預測，提供新鮮人以及有經驗的求職者能更好的進行職涯規劃。

四、研究方法

(一)、使用軟體

- Excel：

Excel 是一個極為實用的工具。它適用於資料處理，並能輔以 VBA 程式，快速且有效地進行資料篩選，提高作業效率。Excel 同樣優秀地支援文件轉換，使得後續的操作更加方便。其直觀的使用介面以及豐富的功能，使得能夠輕鬆地處理大量數據，並進行複雜的數據分析。

- Visual Studio Code：

一個多功能的集成開發環境，其內建支援 Jupyter notebook，並且通過 Python 編程語言的強大支援，使得程式碼的撰寫和測試變得極為方便。使用 VS Code 的 Jupyter 環境，能夠逐區塊測試和運程式碼，這有助於更有效地開發和調試。VS Code 還提供了豐富的擴充插件，支持多種程式語言，使得能夠根據需要自由擴展其功能。

(二)、隨機森林迴歸

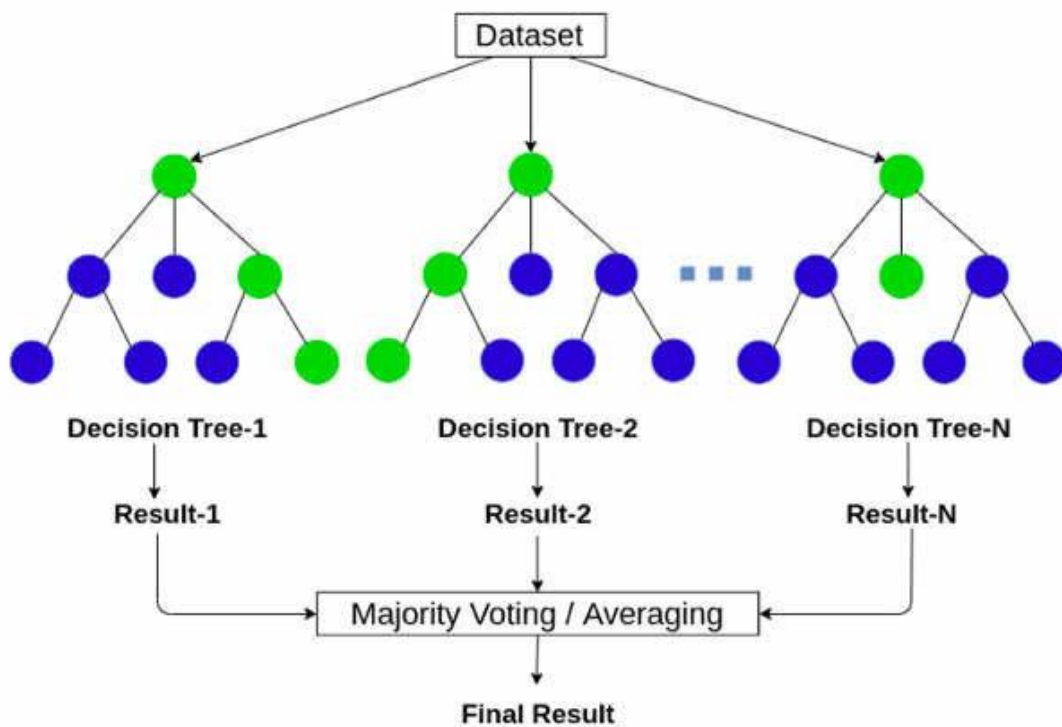
隨機森林迴歸模型結合了多個決策樹的預測，通過隨機性和集成的方法提供了穩健且有效的預測能力，如圖三，其主要的運作方式為：

$$\text{Random Forest} = \text{Bagging} + \text{Decision Tree}$$

透過從資料集中隨機抽取 n 筆資料以建立子樣本，確保每次建模所使用的資料都具有差異性。在每次決策樹的擬合過程中，隨機挑選 k 個特徵，引入隨機性，確保每棵樹都具有多樣性。接著，進行 m 次樹的擬合，每次擬合都針對子樣本和特徵子集建立一棵決策樹。最後，運用迴歸方法對每棵樹進行預測，最終的預測結果是所有樹預測結果的平均值或投票結果，這種集成方法有效地減少過度擬合的風險，同時保持高預測準確性。

隨機森林不僅在擬合模型的方式上具有獨特優勢，在應對不同情境下表現卓越：

- **非線性關係和複雜模式：**隨機森林適合處理資料中複雜的非線性模式和特徵關係，在預測具有多層次結構的資料時，能提供更為精準的結果。。
- **高維度資料：**在擁有很多特徵的資料集上，隨機森林能夠有效的處理，在保持模型穩定性的同時降低過度擬合的風險。
- **大型資料集：**儘管可能不是最快的模型，但隨機森林能在處理大量資料時並行處理，以提高效率，讓模型更具擴展性。
- **混合型特徵：**能處理不同類型的特徵，無需太多特徵工程。
- **特徵重要性評估：**提供了評估特徵重要性的方法，可以看出每個特徵對預測的貢獻程度。



圖三、隨機森林迴歸模型架構示意圖

(三)、模型評估指標

本研究將使用均方根誤差 (Root Mean Squared Error, RMSE) 和決定係數 (Coefficient of Determination, R^2) 作為主要的模型評估指標，以評估隨機森林回歸模型的預測性能。

● 均方根誤差

均方根誤差 (Root Mean Squared Error, RMSE) 是一種廣泛應用的衡量模型預測值與實際值之間的差異的指標，藉此估計預測模型預測目標值的準確度。RMSE 可視為殘差的標準差，即預測誤差的標準差，而殘差為測量值與迴歸線上對應點的距離，因此，RMSE 衡量這些殘差值的分散程度，RMSE 的值越低，則代表預測模型越精準。

以下為 RMSE 的公式，其中 n 為資料中的樣本數， y_i 為實際薪資， \hat{y}_i 是模型預測薪資， $(y_i - \hat{y}_i)^2$ 為每一次預測誤差的平方。

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

● 決定係數

決定係數 (Coefficient of Determination, R^2) 是用來評估統計模型解釋目標變數變異程度的指標，也就是模型對實際數據的解釋能力，即模型所能解釋的目標變異的百分比，可用於評估模型對資料的擬合程度， R^2 介於 0 和 1 之間，越接近 1 表示模型越能解釋目標變數的變異，也代表模型的擬合度較高。

以下為 R^2 的公式，其中 n 為資料中的樣本數， y_i 為實際薪資， \hat{y}_i 是模型預測薪資， \bar{y}_i 為實際薪資的平均數， $(y_i - \hat{y}_i)^2$ 衡量模型預測值和實際值間的差異的平方和， $(y_i - \bar{y}_i)^2$ 衡量目標值的總變異。

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2}$$

五、系統建構

(一)、資料取得和說明

本研究使用之的資料來源於社群媒體 Dcard 上發起的科技業中職務薪資調查，分別為：[科技業薪水分享](#)，其中包含 IC 設計公司、化工化學材料傳統產業、半導體非設計類、系統廠、純軟工程師、設備商等 6 類；[科技業非工程師薪水分享](#)，則涵蓋財務、行銷、專案管理、產品管理、人力資源、策略制定、數據分析、法務、業務、UI/UX 與採購等 11 類。資料欄位分別為：公司名稱、職務、職稱、相關年資(年)、現職年資(年)、月底薪(萬)、總年薪(萬)、Bonus(額外加給)、每月平均工時、加班頻率(天)、爽度、工作量、學校、學系以及最高學歷等 15 項，共計 635 筆資料。

(二)、資料前處理

步驟一：將所有表單根據欄位進行合併，刪除資料填寫量少於 50% 的欄位，如：學校、學系、最高學歷。

步驟二：刪除可能直接推算月薪的欄位，如：總年薪

步驟三：刪除無法參考的數據，如圖四

公司名稱	職務	職級	相關年資	現職年資	月底薪(萬)	Bonus	總年薪(萬)	每月平均工時	每月加班工時
?	?		?	?	?	?	?	?	?

圖四、無意義資料示意圖

步驟四：將職級合併至職務欄位，中間以空格隔開，並刪除職級欄位

步驟五：資料一致性處理，如表一

欄位	處理方式
公司名稱	統一將公司名稱改為英文。
職務	統一將職務改為英文。
相關年資	填入格式有些含有單位，統一刪除單位保留數字。
現職年資	填入格式有些含有單位，統一刪除單位保留數字。
月底薪	填入格式有些含有單位，並且單位不統一，例如四萬表達形式，包含 4 萬，40K 以及 40000，統一該數值，單位統一改為萬並刪除欄位中單位保留數字。
額外加給	填入格式有些含有單位，統一刪除單位保留數字，統一單位為(月)。 公式：IF [額外加給]>100 THEN [額外加給]/[月底薪]*10000 END
每月平均工時	原欄位(每日/月平均工時)無限制每日/月平均工時，統一改為每月平均工時。 公式：IF [每日/月平均工時] < 24 THEN [每日/月平均工時] * 20 END
爽度	原數值中 5 數值為：5 最爽，將此文字改至標題，只保留數字。
工作量	原數值中 5 數值為：5 最重，將此文字改至標題，只保留數字。

表一、資料一致性處理方式

步驟六：將相同的重複數據刪除

步驟七：缺失值處理，如表二

欄位	處理方式
公司名稱	公司名稱若有缺失，刪除該筆資料。
職務	職務若有缺失，刪除該筆資料。
職級	職務若有缺失，則填入空字串。
相關年資	現職年資若有缺失，刪除該筆資料；否則填入現職年資。
現職年資	相關年資若有缺失，刪除該筆資料；否則填入相關年資。
月底薪	月底薪若有缺失，刪除該筆資料。
額外加給	額外加給若有缺失，則填入無缺失值的平均。
每月平均工時	每月平均工時若有缺失，刪除該筆資料。
加班頻率	加班頻率若有缺失，判斷每月平均工時 < 230 ，填入 0 ； > 230 ，填入每月平均工時 $\div 20 - 8$ 。
爽度	爽度若有缺失，刪除該筆資料。
工作量	工作量若有缺失，刪除該筆資料。

表二、缺失值處理方式

註解：爽度與工作量以及加班頻率兩數據的相關性為： -0.4 ，為中等負相關，無法依照相關性填入缺失值。

(三)、資料集內容

通過資料預處理後，本研究最終資料集內容如圖五，共計 368 筆資料，資料欄位說明如下：

- 公司名稱
- 職務
- 相關年資(年)：在此職業工作年資
- 現職年資(年)：現在工作年資

- 月底薪(萬)：每月底薪，單位為萬元
- 額外加給：Bonus
- 每月平均工時：每個月平均工作小時
- 加班頻率(天)：一個星期加班天數
- 爽度：根據工作輕鬆程度，1 為最重，5 為最輕鬆
- 工作量：根據工作量高低，1 為最小，5 為最重

1	公司名稱	職務	相關年資(Y)	現職年資(Y)	月底薪(萬)	Bonus (月)	每月平均工時	加班頻率	爽度(1~5) 5最爽	Loading(5最重)
103	Phison	Engineer	2	2	9.2	6.5	170	2	4	4
104	Phison	Engineer	1	1	9.2	5	180	4	3	3
105	Phison	Digital Designer	2	2	9.6	4	180	3	3	3
106	Phison	Finance Manager	1.5	1.5	9.5	7	160	1	4	3
107	Phison	Finance Manager	1.5	1.5	9.3	4.5	190	4	3	3
108	Phison	Finance Manager	2	2	10	8	160	4	3	4
109	Phison	Project Manager	2	2	6.8	50	140	3	3	4
110	Phison	R&D Engineer	6	2	10	6	160	3	3	3
111	Pegatron	Planner	10	5	6	17	160	1	5	1
112	Panasonic	Software Engineer	1	1	4	7	170	5	3	2
113	Ogilvy	Marketing	2	0	3.5	0	240	5	1	4
114	NVIDIA	R&D Engineer	7	2	23	6	200	4	2	5
115	NVIDIA	Software Engineer	3	1	13	2	240	5	1	5
116	Nuvoton	Software Engineer	1.5	1	8.8	2	160	1	5	2
117	Nuvoton	R&D Engineer	2	2	8.5	6	180	2	1	2
118	NPC	Manager	3	3	5.2	3	180	3	2	3
119	Novatek	Digital Designer Engineer	2.5	1.3	10	10	200	4	2	4
120	Novatek	R&D Engineer	2	2	9	5	180	2	4	2
121	Novatek	Engineer	3	3	10	1	240	2	1	1
122	Novatek	Engineer	2	1	8.5	1	240	5	1	5
123	Novatek	Chief Engineer	6	4	13	15	200	1	4	4
124	Novatek	Advanced Engineer	6	3	9	10	220	4	2	4
125	Novatek	Advanced Engineer	5	2	9	5	200	4	2	4
126	Novatek	Digital RD Engineer	8	8	12	25	220	2	3	4
127	Novatek	Design Verification Director	7	2	10.5	10	160	1	3	4
128	Novatek	Analog IC Engineer	3	3	9.4	8	200	3	2	4
129	Novatek	Digital Designer Engineer	5	5	10	28	220	3	4	4
130	Novatek	Digital Designer Engineer	7	5	11.5	17	220	3	2	4
131	Novatek	R&D Engineer	3	3	9.5	19	180	1	4	4
132	Novatek	R&D Engineer	5	5	10	14	180	2	4	3

圖五、資料集示意圖

(四)、資料庫建構

使用 Python 程式中的 sqlalchemy 套件，將處理好的 csv 資料集的數據儲存至 SQLite，以此來管理數據和確保數據的一致性。此外，SQLite 作為一種輕量級的資料庫，它的便攜性使得數據更容易傳輸和分享，特別是在需要離線處理和分析的場合。在進行薪資預測的過程中，資料將會經歷不斷的擴展，而 SQLite 支援動態調整資料庫的大小，且不會對性能造成重大影響，因此本研究採用 SQLite 作為資料庫。

在初步資料預處理階段已經確保資料的一致性，所以在進行薪資預測模型開發時，為了加速查詢數據、降低資料庫連接的複雜性以及簡化模型訓練的過程，研究中採用未正規化的資料庫。下方為程式碼截圖。

```

from sqlalchemy import create_engine, Column, Integer, String, Float, MetaData, Table, desc
from sqlalchemy.orm import sessionmaker
import pandas as pd
import os

# Database file path
db_file = 'salary_database.db'

# Delete the existing database file if it exists
if os.path.exists(db_file):
    os.remove(db_file)

# Define the database schema
metadata = MetaData()

# Define the table structure
data_table = Table('salary_data', metadata,
    Column('id', Integer, primary_key=True),
    Column('company_name', String),
    Column('position', String),
    Column('related_experience_years', Float),
    Column('current_job_experience_years', Float),
    Column('monthly_salary', Float),
    Column('monthly_bonus', Float),
    Column('average_monthly_working_hours', Integer),
    Column('overtime_frequency', Integer),
    Column('comfort_level', Integer),
    Column('workload', Integer))

# Create a file-based SQLite database
engine = create_engine(f'sqlite:///db_file')
metadata.create_all(engine)

```

```

Session = sessionmaker(bind=engine)
session = Session()

# Load the CSV file
df = pd.read_csv('./rewrite.csv') # Replace with the path to your CSV file

# Function to add a row to the database
def add_row(row):
    insert_statement = data_table.insert().values(
        company_name=row['公司名稱'],
        position=row['職務'],
        related_experience_years=row['相關年資(Y)'],
        current_job_experience_years=row['現職年資(Y)'],
        monthly_salary=row['月底薪(萬)'],
        monthly_bonus=row['Bonus (月)'],
        average_monthly_working_hours=row['每月平均工時'],
        overtime_frequency=row['加班頻率'],
        comfort_level=row['爽度(1~5) 5最爽'],
        workload=row['Loading(5最重)'])
    session.execute(insert_statement)

# Import data into the database
df.apply(add_row, axis=1)
session.commit()

# Querying the last 5 entries from the database
query_result = session.query(data_table).order_by(desc(data_table.c.id)).limit(5).all()
for row in query_result:
    print(row) # This will print out the last five rows from the database

# Close the session
session.close()

```

✓ 1.5s

Python

```

(368, 'MIC', 'Sales', 6.0, 2.0, 4.2, 2.0, 180, 2, 2, 4)
(367, 'ME', 'Sustomer service', 0.0, 1.0, 3.6, 1.0, 230, 0, 5, 0)
(366, 'QUANTA', 'Software Engineer', 5.0, 1.0, 4.5, 5.0, 200, 3, 3, 5)
(365, 'ME', 'VTuber project production', 0.0, 1.0, 3.9, 0.0, 104, 0, 5, 1)
(364, 'Yang Ming Marine Transport Corp', 'Legal', 0.0, 1.0, 4.3, 0.0, 160, 1, 4, 2)

```

(五)、隨機森林模型建構

由於最後總資料筆數只有 368 筆，且欄位之間的相關程度不高，若使用過於複雜的模型架構，反而將導致模型訓練效果不佳且有過度擬合（Overfitting）的情況產生，因此本研究選擇採用隨機森林迴歸（Random Forest Regression）作為主要的分析工具，在參數的設定上，大多選擇預設值，下表三為主要參數設定與結果。

參數	設定值	意義
n_estimators	default=100	表示模型中使用的決策樹的個數。
min_sample_split	default=2	節點再劃分時所需的最小樣本數。
min_sample_leaf	default=1	葉子節點上所需的最小樣本數。
min_weight_fraction_leaf	default=0	葉子節點上的權重最小總和，用於加權數據。
max_features	default=1	進行特徵劃分時考慮的最大特徵數。
random_state	19	控制隨機數生成，確保模型結果的可重現性。
bootstrap	default=True	使用 bootstrap 進行樣本抽取。
<ul style="list-style-type: none">• 均方根誤差（Root Mean Squared Error, RSME）：2.317• 決定係數（Coefficient of Determination, R^2）：0.502		

表三、隨機森林迴歸模型參數設定與結果

在數據量較為有限且特徵之間缺乏高度相關性的情境下，隨機森林迴歸模型的預測成效是中等的，需要進一步優化或者考慮更多特徵以提高其預測精度。

(六)、介面程式開發

使用 Python 取得資料庫中公司名稱與職務欄位中所有 Unique 值，並將數據進行編碼，存成 txt 檔，以便在介面設計中的公司名稱與職務兩欄位的下拉式選單製作，最後透過 Thinker 套件完成介面程式的開發。

- 介面功能：

透過條件確認使用者輸入的值是否符合要求，若不符合要求則會提示錯誤的欄位，並提供 Range 值告知使用者輸入限制，詳細如表四。

欄位	輸入限制
相關年資(Y)	最低不能 < 0 最高不可 > 100
現職年資(Y)	最低不能 < 0 最高不可 > 100
Bonus(月)	最低不能 < 0 最高不可 > 12
加班頻率	最低不能 < 0 最高不可 > 7
每月平均工時	最低不能 < 1 最高不可 > 230
爽度	最低不能 < 1 最高不可 > 5
Loading	最低不能 < 1 最高不可 > 5

表四、使用者輸入限制

- 串接模型：

- 按鈕與預測函數的綁定：

- ◆ 透過 tk.Button 創建一個名為 predict_button 的按鈕，並指定按下按鈕時要執行的函數為 predict_salary()。
- ◆ command=predict_salary 設定了按鈕的點擊事件，當按下按鈕時，它會呼叫 predict_salary 函數。

- 預測函數 (predict_salary) 的實現：

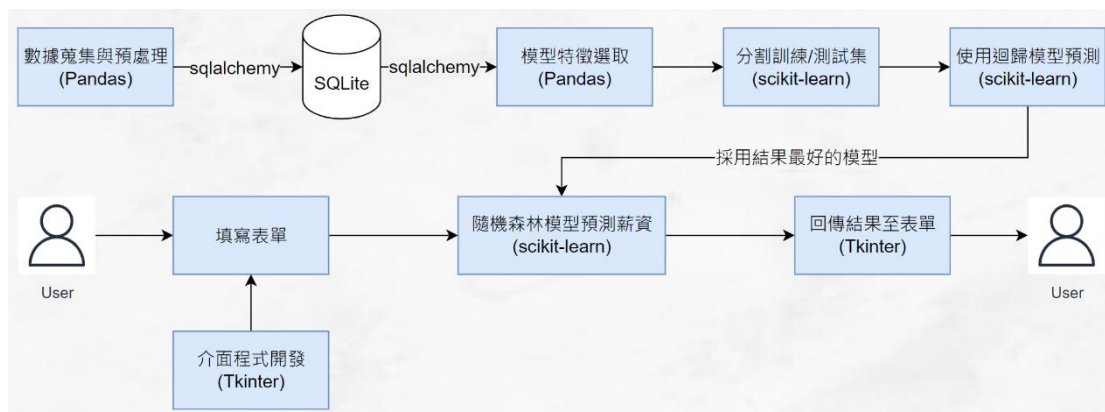
- ◆ 在 predict_salary 函數中，根據使用者在 GUI 輸入的資料，創建一個包含這些資料的 DataFrame。
- ◆ 進行非數值列的標籤編碼(Label Encoding)，確保模型能夠處理文字類別特徵。

- ◆ 使用訓練好的模型（rf_model）對創建的 DataFrame 進行薪資預測。
- ◆ 顯示預測結果在 GUI 上，這裡是透過 result_label.config 設定標籤的文字內容。

```
predicted_salary = rf_model.predict(data)
print("Predicted Salary:", predicted_salary)
result_label.config(text=f"預測薪資: {round(predicted_salary[0], 1)} 萬", fg="black")
```

（七）、系統說明

從圖六架構圖中，本系統將蒐集資料及資料預處理後，挑選出模型需要的特徵，再進行多個迴歸模型的分析，讓使用者透過填寫表單數據，分析工作單位、職稱、相關年資、額外加給以及每月平均工時等數據了解並預測各職位薪資發展現況，將分析出的具體數據結果回傳至表單，顯示在使用者介面給使用者，期望幫助求職者們對未來發展更理解。詳細的系統處理程序可見表五 IPO。



圖六、系統架構圖

#	Function	Input	Process	Output
1	Connect database	salary_database.db	The purpose of <u>Connect database</u> is to create database engine, execute SQL query.	df

			SELECT * FROM salary_data	
2	Process the data	df	The purpose of <u>Process data</u> is to convert columns to numeric, drop missing values, encode non-numeric columns.	df
3	Select features and target	df	The purpose of <u>Select features and target</u> is to select features and target variable for model.	X y
4	Split data	X y	The purpose of <u>Split data</u> is to split dataset into training and testing.	X_train X_test y_train y_test
4	Train model	X_train y_train	The purpose of <u>Train and evaluate model</u> is to train RandomForestRegressor model.	rf_model
5	Predict model	X_test rf_model	The purpose of <u>Predict model</u> is to test the model.	y_pred
6	Evaluate model	y_test y_pred	The purpose of <u>Evaluate model</u> is to calculate RMSE and R^2 .	RMSE R^2

7	Load data for dropdowns	company.txt job_title.txt	The purpose of <u>Load data for dropdowns</u> is to read company and job title data into Tkinter dropdowns.	Company_ job_ title
8	Collect user inputs	1. company_name 2. position 3. related_experience_years 4. current_job_experience_years 6. monthly_bonus 7. average_monthly_ 8. working_hours 9. overtime_frequency 10. comfort_level 11. workload	The purpose of <u>Collect user inputs</u> is to connect the user inputted data and build the DataFrame storage.	data
9	Predict salary	df data rf_model	The purpose of <u>Predict salary</u> is to predict salary using RandomForestRegressor.	predicted_salary
10	Display prediction	predicted_salary	The purpose of <u>Display prediction</u> is to display the predicted salary on the GUI.	

表五、系統 IPO

六、結論與建議

(一)、預期效益

不論是新鮮人或想跨領域的求職者可窺探該預想職業的未來展望；想轉換公司的則可透過自己可接受的加班頻率、現有年資等條件查詢同職業間是否可以得到更高薪資。

- 理解薪資結構：

透過建立薪資預測模型，深入了解職位職稱、工作經驗、加班頻率等會影響薪資結構的各種因素。

- 薪資與工時關聯性分析：

探討薪資與每月平均工時、加班頻率等的相關性，以協助求職者更好預測未來工時和基本薪資、額外加給的關係，更好規劃未來的工時。

- 年資與薪資關聯性分析：

了解深入該職業後未來薪資成長情況，讓求職者更好理解就任該職位後可能面臨的薪資漲幅，以預防深入職位後才發現薪資漲幅不符合預期。

- 協助職涯抉擇：

提供用戶在面對職涯轉換或留在原職位時的決策支援，評估各種選擇對未來職業發展的影響。

(二)、結果展示

詳細之程式碼與成果皆放在 GitHub：https://github.com/School-Project-YZU/DSS_salary_prediction.git，下圖七為有經驗之數據研究員的薪資預測結果。

Field	Value
公司名稱	QCOM
職務	Researcher
相關年資(Y)	10
現職年資(Y)	0
Bonus (月)	1
每月平均工時	123
加班頻率	2
爽度(1~5) 5最爽	1
Loading(5最重)	3
預測薪資	10.2 萬

圖七、薪資預測應用-有經驗之求職者預測結果圖

(三)、問題與改進方向

以下幾點改進方向，將有助於提升系統功能性和使用者體驗，進而提供更全面、實用的資訊給予使用者：

- 提升模型準確率：
擴大樣本數據，確保資料的一致性，特別在額外加給沒有設定單位，數據中會出現不同的呈現形式，透過統一數據，並嘗試其他演算方法，以提高預測的準確率。
- 加強使用者資料蒐集：
提供使用者更全面的服務，透過強化使用者資訊的收集和處理，增加更多欄位數據，如教育程度、系所、技能等，以提供更個人化的薪資預測和職業發展建議。
- 引入職涯規劃功能：
透過給予使用者進修方向和職業發展路徑的建議，協助使用者更全面地規劃自己的職業生涯。
- 定期更新與資料庫管理：

為確保模型能夠應對市場的迅速變化，透過制定正規化資料庫和自動化資料庫更新機制，減少人工確認的時間，在提供最新且最準確的資訊的同時，解放人力。

- 轉變成網頁呈現：

將目前使用 Thinker 在 Local 端 GUI 呈現轉變成透過導入 Flask 等網頁框架，以提供更直觀、便捷的網頁介面，讓系統更容易部署與維護。

七、人員分配

學號	姓名	工作內容
1091603	池昀堯	簡報製作、文件撰寫、資料處理
1091606	曾暄雅	文件撰寫、資料整合及處理、程式修改
1091613	林芸晨	文件撰寫、介面製作、資料整合
1091728	蔡芊葳	文件撰寫、資料處理、資料庫與模型程式
1091752	黃湘樺	簡報製作、文件撰寫、資料處理

八、參考資料

- [1] 104 職場力, 2022 年員工 C.E.O. 工作價值認知調查報告, 上網日期 2022 年 09 月 14 日, 檢自: <https://blog.104.com.tw/2022-ceo-104/>
- [2] 洪躍峰、王靖瑩、楊璨儀、張哲偉, (2014)。影響台灣薪資的因素探討。檢自: <http://chur.chu.edu.tw/bitstream/987654321/42635/2>
- [3] 廖宜川, (2020)。基於人力銀行之台灣地區薪資預測模型。檢自: <https://hdl.handle.net/11296/4j4d25>
- [4] Chung-Yi, ML 入門 (十七) 隨機森林(Random Forest), 上網日期 2019 年 09 月 28 日, 檢自: <https://medium.com/chung-yi/ml%E5%85%A5%E9%96%80-%E5%8D%81%E4%B8%83-%E9%9A%A8%E6%A9%9F%E6%A3%AE%E6%9E%97-random-forest-6afc24871857>
- [5] Ha, Vo Thanh. 2023. “Experimental Study on Remaining Useful Life Prediction of Lithium-Ion Batteries Based on Three Regressions Models for Electric Vehicle Applications.” Preprints. <https://doi.org/10.20944/preprints202306.0999.v1>.
- [6] Wikipedia contributors. 2022. “均方根誤差.” Wikipedia, The Free Encyclopedia. 2022.

<https://zh.wikipedia.org/w/index.php?title=%E5%9D%87%E6%96%B9%E6%A0%B9%E8%AF%AF%E5%B7%AE&oldid=73865911>.

- [7] Wikipedia contributors. 2023. “決定係數.” Wikipedia, The Free Encyclopedia. 2023.

<https://zh.wikipedia.org/w/index.php?title=%E5%86%B3%E5%AE%9A%E7%B3%BB%E6%95%B0&oldid=77501730>.

- [8] 怨工作量與薪水不成正比！1 成 5 新鮮人上工不開心、月薪平均近 3 萬，上網日期 2023 年 08 月 23 日，檢自：

<https://www.nownews.com/news/6236995>