BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC PHENIKAA

BÁO CÁO HỌC PHẦN LẬP TRÌNH PHÂN TÍCH DỮ LIỆU VỚI PYTHON

Nhóm 19
Đề tài: Phân tích yếu tố quan trọng ảnh hưởng đến mức lương trung bình của ngành công nghệ

Thành viên nhóm

Nguyễn Thái Sơn	**23010196**	**ICT_VJ-1**

GVHD: Nguyễn Anh Tuấn

Hà Nội – Năm 2025

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC PHENIKAA

BÁO CÁO HỌC PHẦN LẬP TRÌNH PHÂN TÍCH DỮ LIỆU VỚI PYTHON

Nhóm 19
Đề tài: Phân tích yếu tố quan trọng ảnh hưởng đến mức lương trung bình của ngành công nghệ

Thành viên nhóm

Nguyễn Thái Sơn	**23010196**	**ICT_VJ-1**

GVHD: Nguyễn Anh Tuấn

Hà Nội – Năm 2025

MỤC LỤC

DANH MỤC HÌNH ẢNH

MỞ ĐẦU

Trong bối cảnh cuộc cách mạng công nghiệp 4.0 đang diễn ra mạnh mẽ trên toàn thế giới, Việt Nam cũng đang đối mặt với những cơ hội và thách thức to lớn trên nhiều lĩnh vực như kinh tế, kỹ thuật, y tế,… đặc biệt là trong lĩnh vực công nghệ thông tin. Sự phát triển nhanh chóng của công nghệ thông tin đã trở thành xu hướng tất yếu trong đời sống hiện đại, đòi hỏi các kỹ sư và sinh viên ngành công nghệ thông tin phải không ngừng trau dồi kiến thức chuyên môn như ngôn ngữ lập trình, cơ sở dữ liệu, lập trình hướng đối tượng, khai phá và phân tích dữ liệu,… Đồng thời, họ cũng cần có khả năng ứng dụng hiệu quả các kiến thức đó vào thực tiễn để bắt kịp với nhịp độ phát triển của thời đại số.

Xuất phát từ thực tiễn đó, nhóm chúng em lựa chọn thực hiện đề tài **"Phân tích yếu tố quan trọng ảnh hưởng đến mức lương trung bình của ngành công nghệ"**. Đề tài nhằm mục tiêu làm rõ những yếu tố then chốt có ảnh hưởng đến mức thu nhập của người lao động trong lĩnh vực công nghệ thông tin – ngành nghề đang giữ vai trò trọng yếu trong quá trình chuyển đổi số và phát triển kinh tế tri thức tại Việt Nam. Việc nhận diện và đánh giá các yếu tố này không chỉ giúp sinh viên có định hướng nghề nghiệp rõ ràng, mà còn mang lại cái nhìn khách quan, khoa học cho doanh nghiệp và xã hội trong việc phát triển nguồn nhân lực chất lượng cao.

Trong quá trình thực hiện đề tài, nhóm đã nhận được sự hướng dẫn tận tâm và đầy trách nhiệm của **Thầy Nguyễn Anh Tuấn** – giảng viên phụ trách học phần ***Lập trình phân tích dữ liệu với Python*.** Thầy đã truyền đạt cho chúng em những kiến thức hữu ích, giải đáp những vướng mắc chuyên môn, đồng thời hỗ trợ chúng em rèn luyện kỹ năng phân tích và xử lý dữ liệu một cách thực tiễn và hiệu quả.

Chúng em kỳ vọng rằng, đề tài này sau khi hoàn thiện sẽ trở thành một sản phẩm học tập có giá trị, đồng thời đóng góp một phần thiết thực vào việc giải quyết các vấn đề thực tế liên quan đến nhân lực trong ngành công nghệ thông tin.

Nhóm em xin trân trọng cảm ơn!

NỘI DUNG

1. Tóm tắt

Báo cáo này trình bày toàn bộ quá trình phân tích, xây dựng và đánh giá các mô hình học máy nhằm dự đoán mức lương trung bình trong ngành công nghệ, đồng thời xác định những yếu tố có ảnh hưởng lớn đến thu nhập của người lao động trong lĩnh vực này. Dữ liệu được sử dụng trong bài toán bao gồm nhiều đặc trưng đa dạng như chức danh công việc, cấp bậc (seniority), hình thức làm việc (remote/on-site), quốc gia làm việc, năm làm việc, cũng như các yếu tố liên quan đến kỹ năng và vị trí địa lý. Dữ liệu đã được xử lý, mã hóa và chuẩn hóa để đưa vào mô hình huấn luyện.

Mục tiêu chính của bài tập lớn là xây dựng các mô hình dự báo hiệu quả, so sánh hiệu năng giữa các thuật toán hồi quy phổ biến, từ đó lựa chọn được mô hình tối ưu cho bài toán thực tế. Các mô hình được áp dụng bao gồm: **Linear Regression, Decision Tree Regressor, Random Forest Regressor, XGBoost Regressor** và **K-Nearest Neighbors (KNN) Regressor**. Mỗi mô hình được chạy lặp lại 10 lần với các giá trị random_state khác nhau nhằm đánh giá tính ổn định và khả năng tổng quát hóa. Các chỉ số đánh giá hiệu suất chính bao gồm: **R-squared (R^2), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE)**, cùng với thời gian huấn luyện của từng mô hình.

Kết quả thực nghiệm cho thấy **XGBoost là mô hình có hiệu suất dự báo tốt nhất**, với R^2 cao nhất (trung bình ~0.46) và sai số thấp nhất (RMSE ~0.37), đồng thời duy trì được độ ổn định cao qua các lần chạy. **Random Forest** cũng đạt hiệu quả khá tốt, xếp sau XGBoost, trong khi các mô hình đơn giản như **Linear Regression** và **Decision Tree** cho kết quả trung bình và dao động nhiều hơn. **KNN** là mô hình hoạt động kém nhất trong bối cảnh dữ liệu này, cho thấy sự hạn

chế trong việc xử lý các tập dữ liệu có nhiều đặc trưng không chuẩn hóa và có mối quan hệ phi tuyến tính phức tạp.

Các biểu đồ và bảng thống kê được sử dụng để minh họa trực quan hiệu suất của các mô hình, thể hiện rõ sự khác biệt giữa các chỉ số R^2, RMSE, MAE cũng như sự ổn định trong từng mô hình qua 10 lần thử nghiệm. Bên cạnh đó, thời gian huấn luyện của các mô hình cũng được ghi nhận để cân nhắc yếu tố hiệu quả thực tiễn trong các ứng dụng triển khai thực tế.

Từ các kết quả thu được, báo cáo đã phân tích vai trò của từng yếu tố đầu vào đối với mức lương, đồng thời khẳng định tầm quan trọng của việc lựa chọn thuật toán phù hợp, quy trình tiền xử lý dữ liệu và kỹ thuật đánh giá đa chiều trong các bài toán dự báo

sử dụng học máy. Bài tập lớn không chỉ mang lại kết quả dự báo có độ chính xác cao mà còn cung cấp các góc nhìn sâu sắc về dữ liệu ngành công nghệ, góp phần hỗ trợ việc ra quyết định trong các bài toán nhân sự và hoạch định chiến lược lương thưởng trong tương lai.

2. Giới thiệu

Môn học *Lập trình phân tích dữ liệu* trang bị cho sinh viên nền tảng kiến thức vững chắc về cách thu thập, xử lý và phân tích dữ liệu bằng các công cụ lập trình hiện đại.

Trong thời đại mà dữ liệu trở thành một loại "tài nguyên chiến lược", khả năng trích xuất thông tin giá trị từ dữ liệu không chỉ là yêu cầu chuyên môn mà còn là lợi thế cạnh tranh quan trọng trong nhiều lĩnh vực. Môn học không chỉ rèn luyện kỹ năng kỹ thuật mà còn giúp sinh viên phát triển tư duy logic và năng lực ra quyết định dựa trên dữ liệu thực tế.

Trong khuôn khổ bài tập lớn của học phần, nhóm em lựa chọn đề tài **"Phân tích yếu tố quan trọng ảnh hưởng đến mức lương trung bình của ngành công**

nghệ”, với mục tiêu tìm hiểu những biến số nào có tác động đáng kể đến thu nhập của người lao động trong ngành. Vấn đề này có ý nghĩa thực tiễn rõ rệt trong bối cảnh ngành công nghệ thông tin đang phát triển nhanh và nhu cầu nhân lực chất lượng cao ngày càng tăng. Việc nắm bắt được mối quan hệ giữa các yếu tố như kinh nghiệm, kỹ năng, học vấn và mức lương sẽ hỗ trợ sinh viên định hướng nghề nghiệp hiệu quả hơn, đồng thời giúp doanh nghiệp có thêm căn cứ trong hoạch định nhân sự.

Trong bài tập lớn này, nhóm sử dụng ngôn ngữ lập trình **Python** cùng với các thư viện như **Pandas**, **Matplotlib**, **Seaborn** và đặc biệt là **Scikit-learn** để xây dựng mô hình **phân tích hồi quy tuyến tính**. Đây là phương pháp chủ đạo giúp xác định và đo lường mức độ ảnh hưởng của từng yếu tố đến biến phụ thuộc là *mức lương trung bình*. Qua đó, nhóm không chỉ áp dụng được kiến thức lý thuyết vào thực tiễn mà còn có cơ hội tiếp cận với quy trình phân tích dữ liệu hoàn chỉnh, từ thu thập đến trực quan hóa và xây dựng mô hình dự đoán.

3. Kiến thức nền tảng.

Để thực hiện đề tài “Phân tích yếu tố quan trọng ảnh hưởng đến mức lương trung bình của ngành công nghệ”, nhóm đã vận dụng nhiều kiến thức cốt lõi thuộc lĩnh vực phân tích dữ liệu, học máy cơ bản và lập trình với Python. Trong phần này, chúng em trình bày các thuật toán, kỹ thuật và công cụ đã sử dụng, đồng thời nêu rõ cơ sở lý thuyết và tài liệu tham khảo đi kèm.

3.1. Hồi quy tuyến tính (Linear Regression)

Hồi quy tuyến tính là một trong những phương pháp cơ bản nhất trong phân tích dữ liệu định lượng, được sử dụng để mô hình hóa mối quan hệ giữa một biến phụ thuộc (target) và một hoặc nhiều biến độc lập (features). Trong đề tài này, biến phụ thuộc là **mức lương trung bình**, và các biến độc lập có thể là **số năm kinh nghiệm**, **vị trí công việc**, **trình độ học vấn**, **ngôn ngữ lập trình**, v.v.

Công thức mô hình tuyến tính bội: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$

Trong đó:

- y: biến phụ thuộc (mức lương).
- $x_1, x_2, \ldots, x_n$: các biến độc lập.
- β_0: hệ số chặn.
- $\beta_1, \beta_2, \ldots, \beta_n$: hệ số hồi quy.
- ε: sai số (residuals).

Thông qua hồi quy tuyến tính, chúng ta có thể:

- Xác định mức độ ảnh hưởng của từng yếu tố đến mức lương.
- Dự đoán lương dựa trên thông tin đầu vào.
- Phân tích độ chính xác của mô hình dựa trên các chỉ số như R^2, MSE, p-value.

3.2. Ngôn ngữ lập trình và thư viện phân tích dữ liệu

Để triển khai mô hình, chúng em sử dụng **Python**, một ngôn ngữ lập trình mạnh mẽ, phổ biến trong lĩnh vực khoa học dữ liệu, nhờ vào cú pháp đơn giản và hệ sinh thái thư viện phong phú.

3.2.1. NumPy

Numpy là thư viện quan trọng trong Python, chuyên dùng cho các phép toán số học và xử lý mảng (arrays) đa chiều hiệu quả. Numpy cung cấp các hàm đại số tuyến tính, hàm thống kê và công cụ tạo mảng, giúp việc tính toán toán học trở nên nhanh chóng và chính xác hơn trong xử lý dữ liệu. Đây là nền tảng để các thư viện khác như Pandas và

Scikit-learn phát triển.

3.2.2. Pandas

Pandas là thư viện mạnh mẽ dùng để đọc, xử lý và phân tích dữ liệu dạng bảng như file CSV, Excel hoặc dữ liệu từ cơ sở dữ liệu. Nó cung cấp các cấu trúc dữ liệu linh hoạt như DataFrame và Series, hỗ trợ thao tác lọc, sắp xếp, nhóm và tiền xử lý dữ liệu rất tiện lợi. Đây là công cụ không thể thiếu trong quy trình chuẩn bị dữ liệu cho phân tích và mô hình hóa.

3.2.3. Matplotlib và Seaborn

Matplotlib là thư viện đồ họa phổ biến dùng để tạo biểu đồ, hình ảnh trực quan giúp minh họa dữ liệu và kết quả phân tích một cách trực quan. Seaborn là thư viện xây dựng trên Matplotlib, cung cấp các kiểu biểu đồ đẹp mắt và dễ dùng hơn, phù hợp cho việc phân tích thống kê và trình bày báo cáo.

3.2.4. Scikit-learn

Scikit-learn là thư viện hàng đầu trong lĩnh vực học máy (Machine Learning) với rất nhiều thuật toán và công cụ cho các bài toán như hồi quy (Regression), phân loại (Classification), phân cụm (Clustering), dự báo chuỗi thời gian (Time-series Prediction),
v.v. Thư viện này giúp xây dựng, huấn luyện và đánh giá các mô hình một cách dễ dàng và hiệu quả.

3.3. Chuẩn bị và mã hóa dữ liệu (Data Preprocessing)

3.4. Đánh giá mô hình

Mô hình hồi quy được đánh giá dựa trên các chỉ số:

- **R^2 (Hệ số xác định):** cho biết phần trăm biến động của biến phụ thuộc được giải thích bởi mô hình.
- **MSE (Mean Squared Error):** độ sai lệch trung bình bình phương.

- **p-value:** giúp kiểm định ý nghĩa thống kê của từng biến độc lập.

Các chỉ số này giúp chúng em hiểu được mức độ chính xác và độ tin cậy của mô hình trong việc phân tích yếu tố ảnh hưởng đến lương.

4. Thu thập và xử lí dữ liệu

4.1 Thu thập dữ liệu

Bộ dữ liệu được sử dụng trong bài tập lớn này có tên là **"global_tech_salary"**, được thu thập từ trang web Kaggle, một nền tảng chia sẻ dữ liệu và cuộc thi khoa học dữ liệu nổi tiếng. Bộ dữ liệu chứa thông tin về mức lương của nhiều vai trò khác nhau trong lĩnh vực khoa học dữ liệu, được tổng hợp từ nhiều quốc gia và khu vực trên thế giới. Các thông tin bao gồm năm làm việc (work_year), mức độ kinh nghiệm (experience_level), loại hợp đồng (employment_type), chức danh công việc (job_title), lương theo đơn vị tiền gốc (salary) và lương quy đổi sang USD (salary_in_usd), địa điểm sinh sống và làm việc, tỷ lệ làm việc từ xa và quy mô công ty.

4.2 Tổng quan về các trường dữ liệu

Bộ dữ liệu bao gồm các cột chính sau:

- work_year: Năm dữ liệu tiền lương được ghi nhận (ví dụ: 2023, 2024).
- experience_level: Mức độ kinh nghiệm (Entry-level, Mid-level, Senior-level, Executive-level).
- employment_type: Loại hình làm việc (ví dụ: FT – Full-time).
- job_title: Tên công việc (Data Scientist, Machine Learning Engineer, v.v.).
- salary, salary_currency, salary_in_usd: Lương theo tiền gốc và quy đổi sang USD.
- employee_residence: Quốc gia nơi nhân viên sinh sống.

- remote_ratio: Mức độ làm việc từ xa (0 – tại chỗ, 50 – hybrid, 100 – hoàn toàn từ

  xa).

- company_location: Quốc gia nơi công ty đặt trụ sở.
- company_size: Quy mô công ty (S – nhỏ, M – vừa, L-

  lớn). Dưới đây là một số dòng đầu tiên của tập dữ liệu:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	2023	MI	FT	Data Analyst	165000	USD	165000	AU	0	AU	M
1	2023	MI	FT	Data Analyst	70000	USD	70000	US	100	US	M
2	2024	MI	FT	Machine Learning Engineer	85000	EUR	94444	IE	100	IE	M
3	2024	SE	FT	Data Scientist	92700	USD	92700	US	0	US	M
4	2023	MI	FT	Research Engineer	150000	USD	150000	US	0	US	M

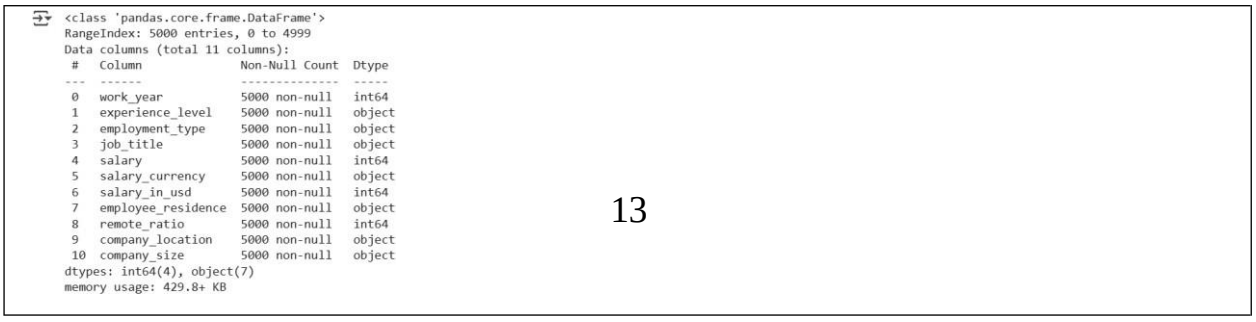
Hình 1. Một số dòng đầu tiên của tập dữ liệu

4.3. Tiền xử lý dữ liệu

Sau khi thu thập được tập dữ liệu thô, nhóm tiến hành các bước tiền xử lý nhằm đảm bảo chất lượng và độ chính xác trước khi đưa vào phân tích và xây dựng mô hình dự đoán. Các bước được thực hiện bao gồm: kiểm tra dữ liệu trùng lặp và thiếu, xử lý ngoại lệ, mã hóa biến phân loại, chuẩn hóa dữ liệu số và xử lý biến mục tiêu.

4.3.1. Kiểm tra dữ liệu ban đầu

Trước tiên, dữ liệu được kiểm tra để xác định kích thước, kiểu dữ liệu của từng cột và tình trạng thiếu dữ liệu. Kết quả cho thấy bộ dữ liệu bao gồm hơn 3700 dòng và không xuất hiện giá trị thiếu trong các cột quan trọng.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5000 entries, 0 to 4999
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   work_year           5000 non-null   int64 
 1   experience_level    5000 non-null   object
 2   employment_type     5000 non-null   object
 3   job_title           5000 non-null   object
 4   salary              5000 non-null   int64 
 5   salary_currency     5000 non-null   object
 6   salary_in_usd       5000 non-null   int64 
 7   employee_residence  5000 non-null   object
 8   remote_ratio        5000 non-null   int64 
 9   company_location    5000 non-null   object
 10  company_size        5000 non-null   object
dtypes: int64(4), object(7)
memory usage: 429.8+ KB
```

Hình 2. Thông tin của tệp dữ liệu

4.3.2. Xử lý trùng lặp

Việc loại bỏ các dòng dữ liệu trùng lặp được đánh giá là bước thiết yếu nhằm tránh làm sai lệch kết quả phân tích và huấn luyện mô hình học máy. Dữ liệu trùng có thể khiến mô hình học quá mức trên những mẫu lặp lại, dẫn đến hiện tượng overfitting - một vấn đề làm giảm khả năng tổng quát hóa của mô hình khi áp dụng trên dữ liệu mới. Ngoài ra, dữ liệu trùng còn gây lãng phí tài nguyên tính toán, làm tăng thời gian huấn luyện mô hình mà không cung cấp thêm thông tin mới. Vì vậy, nhóm đã tiến hành rà soát và loại bỏ toàn bộ các bản ghi trùng, giúp làm sạch và cân bằng tập dữ liệu.

4.3.3 Xử lý ngoại lệ

Một vấn đề thường gặp trong phân tích dữ liệu là sự tồn tại của các giá trị ngoại lệ (outliers), đặc biệt là trong biến lương (salary_in_usd), có thể gây ảnh hưởng tiêu cực đến kết quả phân tích và hiệu quả của mô hình dự báo. Để giảm thiểu tác động này, nhóm đã áp dụng phương pháp lọc giá trị ngoại lệ bằng cách giữ lại những bản ghi có mức lương nằm trong khoảng phân vị từ 1% đến 99%.

```
Ngưỡng dưới (1%): 28825.9
Ngưỡng trên (99%): 336804.9999999998
Kích thước dữ liệu sau khi lọc: (3778, 11)
```

Hình 3. Kết quả xử lý ngoại lệ

Qua đó, các mức lương bất thường quá thấp hoặc quá cao đã được loại trừ, giúp phân phối dữ liệu trở nên đồng nhất và hợp lý hơn cho các bước xử lý tiếp theo.

4.3.4. Mã hóa dữ liệu dạng phân loại

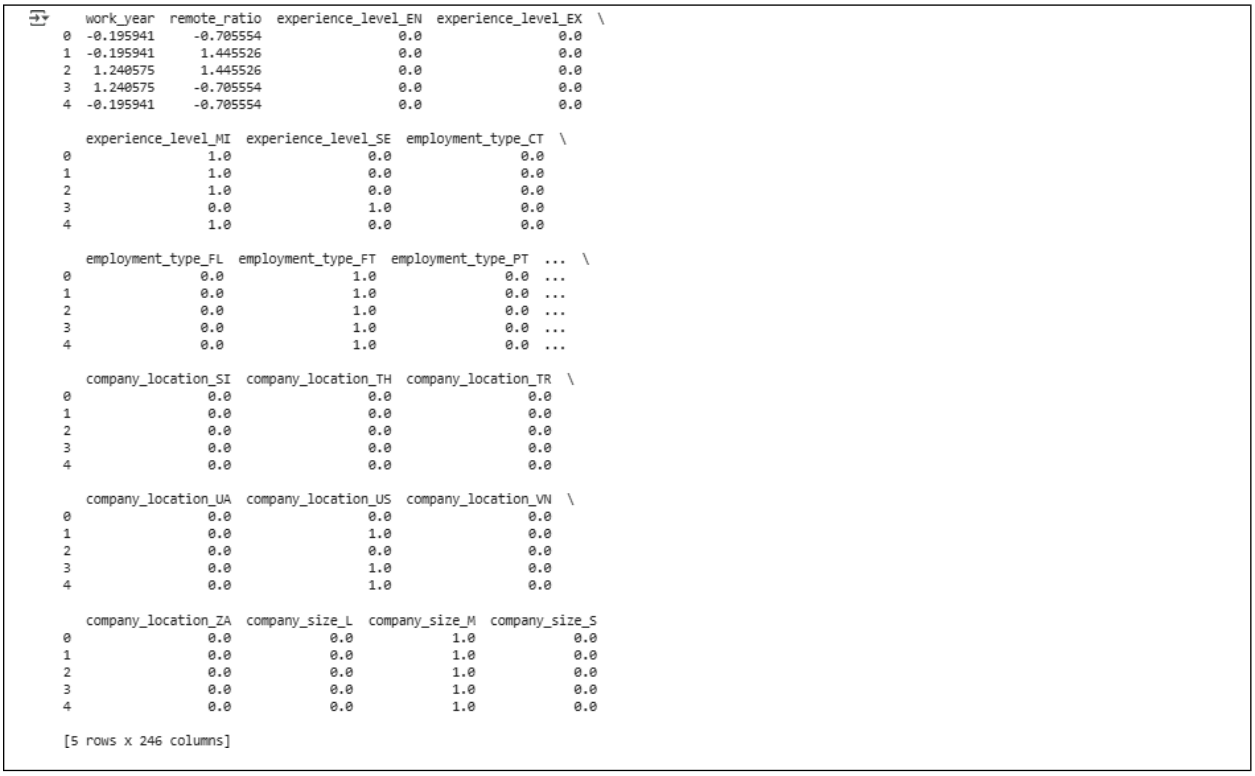
Các biến dạng phân loại (categorical variables) như “job_title”, “experience_level”, “employment_type”, “employee_residence”, “company_location” và “company_size” không thể trực tiếp sử dụng trong các thuật toán học máy, do đó nhóm đã tiến hành chuyển đổi chúng sang dạng số thông qua kỹ thuật mã hóa one-hot (one-hot encoding).

Đặc biệt, để tránh việc làm tăng quá mức số chiều dữ liệu và hạn chế hiện tượng thưa thớt (sparsity), nhóm đã thực hiện gom các giá trị ít phổ biến trong cột “job_title” thành một nhóm chung có tên “Other”. Việc này giúp cân bằng dữ liệu và nâng cao hiệu quả của quá trình huấn luyện mô hình.

4.3.5. Chuẩn hóa dữ liệu số

Cuối cùng, nhóm thực hiện chuẩn hóa các biến số liên tục như “work_year” và “remote_ratio” bằng phương pháp chuẩn hóa z-score (standardization). Việc chuẩn hóa này nhằm đưa các biến về cùng thang đo, tránh hiện tượng một số đặc trưng có giá trị lớn chi phối mô hình, đồng thời giúp tăng tốc độ hội tụ và hiệu quả huấn luyện các thuật toán học máy về sau.

Sau tất cả các bước, nhóm thu được kết quả của tệp dữ liệu như sau:



	work_year	remote_ratio	experience_level_EN	experience_level_EX	\
0	-0.195941	-0.705554	0.0	0.0	
1	-0.195941	1.445526	0.0	0.0	
2	1.240575	1.445526	0.0	0.0	
3	1.240575	-0.705554	0.0	0.0	
4	-0.195941	-0.705554	0.0	0.0	

	experience_level_MI	experience_level_SE	employment_type_CT	\
0	1.0	0.0	0.0	
1	1.0	0.0	0.0	
2	1.0	0.0	0.0	
3	0.0	1.0	0.0	
4	1.0	0.0	0.0	

	employment_type_FL	employment_type_FT	employment_type_PT	...	\
0	0.0	1.0	0.0	...	
1	0.0	1.0	0.0	...	
2	0.0	1.0	0.0	...	
3	0.0	1.0	0.0	...	
4	0.0	1.0	0.0	...	

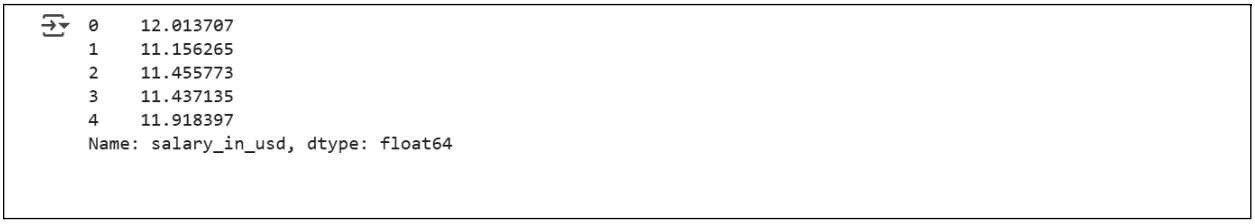
	company_location_SI	company_location_TH	company_location_TR	\
0	0.0	0.0	0.0	
1	0.0	0.0	0.0	
2	0.0	0.0	0.0	
3	0.0	0.0	0.0	
4	0.0	0.0	0.0	

	company_location_UA	company_location_US	company_location_VN	\
0	0.0	0.0	0.0	
1	0.0	1.0	0.0	
2	0.0	0.0	0.0	
3	0.0	1.0	0.0	
4	0.0	1.0	0.0	

	company_location_ZA	company_size_L	company_size_M	company_size_S
0	0.0	0.0	1.0	0.0
1	0.0	0.0	1.0	0.0
2	0.0	0.0	1.0	0.0
3	0.0	0.0	1.0	0.0
4	0.0	0.0	1.0	0.0

[5 rows x 246 columns]

Hình 4. Các dòng đầu của dữ liệu features đã xử lý



```
0    12.013707
1    11.156265
2    11.455773
3    11.437135
4    11.918397
Name: salary_in_usd, dtype: float64
```

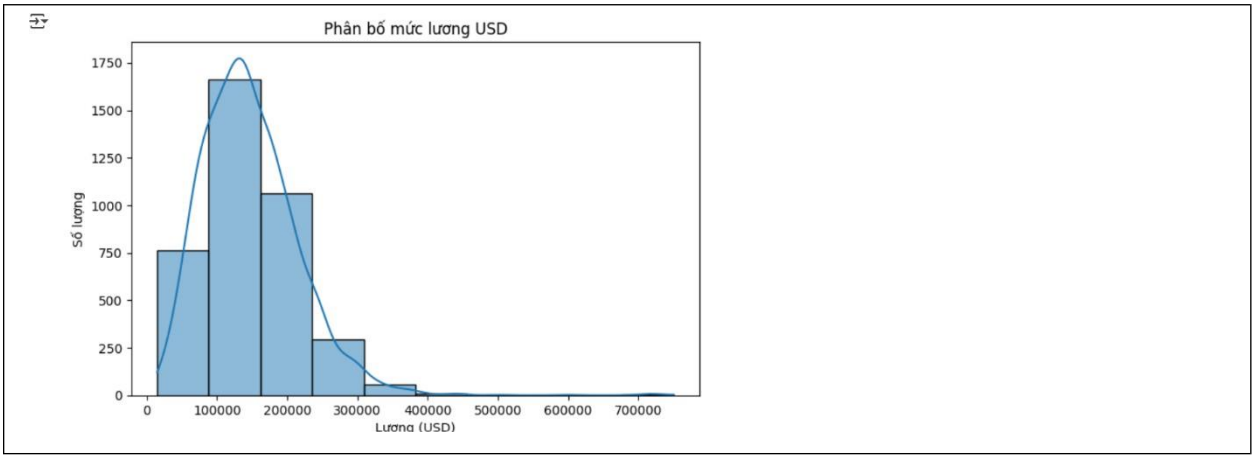
Hình 5. Các dòng đầu của biến mục tiêu

5. Phân tích dữ liệu

Phần này trình bày kết quả khai phá dữ liệu, trực quan hóa và phân tích các đặc trưng quan trọng từ bộ dữ liệu **global_tech_salary**. Qua các biểu đồ và bảng số liệu, nhóm em sẽ mô tả các xu hướng, đặc điểm nổi bật, cũng như đưa ra các nhận xét chuyên sâu nhằm hỗ trợ quá trình xây dựng mô hình dự báo lương.

5.1. Phân bố mức lương theo USD

Để hiểu tổng quan về mức lương trong bộ dữ liệu, nhóm đã vẽ biểu đồ phân bố histogram với đường mật độ ước lượng (KDE) cho biến salary_in_usd.



Hình 6. Biểu đồ phân bố mức lương

Qua biểu đồ ta thấy được:

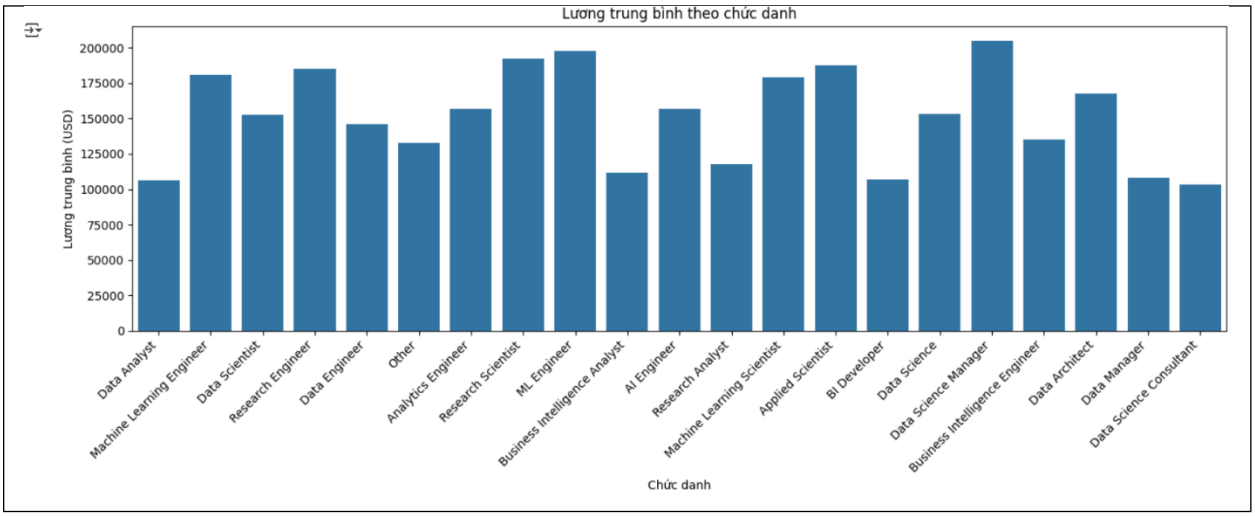
- Biểu đồ có hình dạng lệch phải (skewed right), nghĩa là đa số người lao động nhận mức lương từ thấp đến trung bình, trong khi một số ít người hưởng mức lương rất cao tạo thành đuôi dài bên phải.
- Khoảng mức lương phổ biến tập trung chủ yếu vào vùng 100,000 – 150,000 USD,

  đây là mức lương trung bình của đa số các vị trí công việc.
- Các mức lương trên 300,000 USD rất hiếm gặp, cho thấy mức thu nhập này thuộc nhóm cao cấp và không phổ biến rộng rãi.
- Một số giá trị ngoại lệ vượt quá 500,000 USD tồn tại nhưng không ảnh hưởng đáng kể đến xu hướng chung của dữ liệu.

Biểu đồ phân bố này giúp nhóm nhận biết được đặc điểm phân phối của biến mục tiêu, từ đó lựa chọn phương pháp xử lý phù hợp khi xây dựng mô hình.

5.2. Mức lương trung bình theo chức danh công việc

Nhóm tiếp tục phân tích mối quan hệ giữa chức danh công việc và mức lương trung bình bằng biểu đồ cột.



Hình 7. Mức lương trung bình theo chức danh công việc

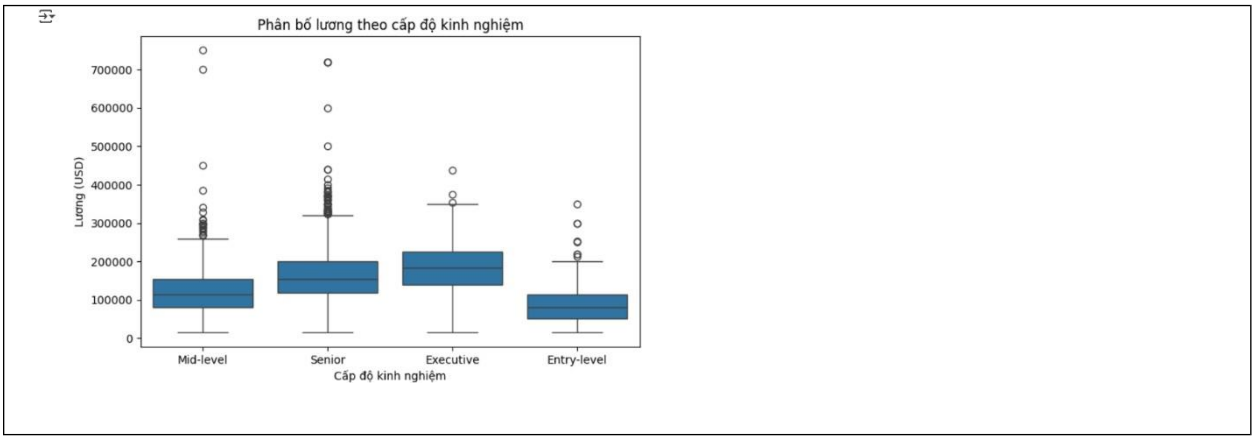
Qua biểu đồ, ta thấy được:

- Các vị trí quản lý hoặc chuyên môn cao như *Data Science Manager* có mức lương trung bình cao nhất, trên 200,000 USD.
- Các chức danh như *Machine Learning Engineer*, *Research Scientist*, *BI Developer*, *Research Engineer* cũng có mức lương khá cao, dao động trong khoảng 185,000–195,000 USD.
- Nhóm chức danh phổ biến như *Data Scientist*, *AI Engineer*, *Applied Scientist* có mức lương trung bình nằm trong khoảng 150,000–180,000 USD.
- Một số chức danh có mức lương trung bình thấp hơn như *Data Analyst*, *Business Intelligence Analyst* dưới 120,000 USD.
- Có sự chênh lệch đáng kể về mức lương giữa các chức danh, phản ánh rõ ràng sự khác biệt về yêu cầu kỹ năng, trách nhiệm và kinh nghiệm công việc.

Biểu đồ này giúp nhóm hiểu rõ hơn về phân tầng thu nhập theo vai trò, từ đó có thể cân nhắc phân nhóm dữ liệu hoặc xây dựng các biến đặc trưng phù hợp cho mô hình.

5.3. Mức lương theo cấp độ kinh nghiệm

Phân tích mức lương theo từng cấp bậc kinh nghiệm giúp đánh giá ảnh hưởng của thời gian làm việc và chuyên môn đến thu nhập.



Hình 8. Mức lương theo cấp độ kinh nghiệm

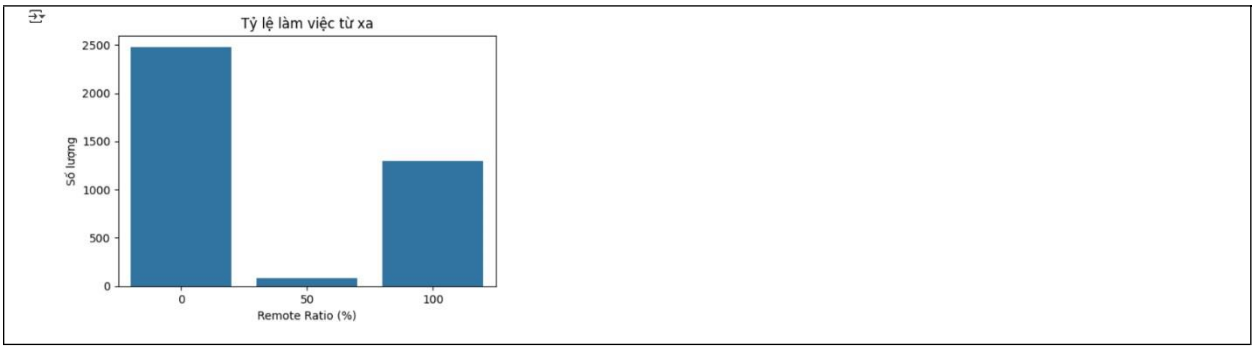
Qua biểu đồ ta thấy được:

- Xu hướng chung là mức lương trung vị tăng dần từ *Entry-level* → *Mid-level* → *Senior* → *Executive*, phản ánh quá trình tích lũy kinh nghiệm và thăng tiến nghề nghiệp.
- *Entry-level* có mức lương thấp nhất với biên độ thu nhập hẹp, phù hợp với người mới vào nghề.
- *Senior* và *Executive* có độ phân tán lương lớn, nhiều điểm ngoại lệ thể hiện sự đa dạng về mức thu nhập trong cùng một cấp bậc, có thể do sự khác biệt về kỹ năng chuyên môn, ngành nghề hoặc vị trí địa lý.
- *Executive* có trung vị lương đôi khi thấp hơn *Senior* trong bộ dữ liệu này, điều này có thể do số lượng mẫu nhỏ hoặc các đặc thù ngành nghề riêng biệt.

Phân tích này là cơ sở để nhóm đánh giá tầm quan trọng của biến cấp độ kinh nghiệm trong mô hình dự báo và lựa chọn kỹ thuật xử lý phù hợp.

5.4. Phân bố hình thức làm việc từ xa

Biểu đồ đếm (countplot) được sử dụng để khảo sát tỷ lệ phần trăm làm việc từ xa trong bộ dữ liệu.



Hình 9. Tỉ lệ việc làm từ xa

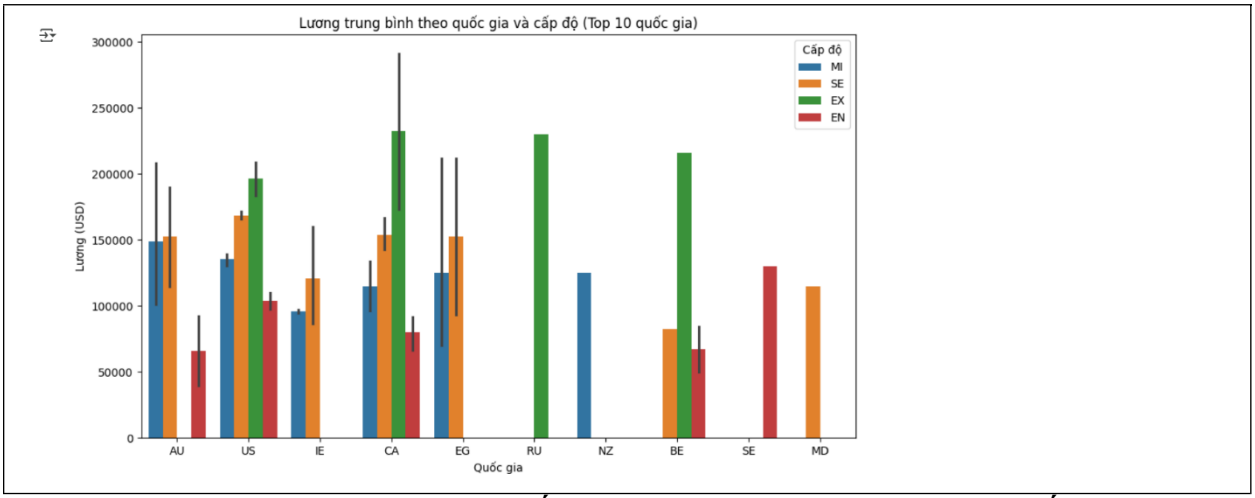
Từ biểu đồ trên, có thể dễ dàng nhận ra:

- Đa số nhân viên (~2,500 người) làm việc hoàn toàn tại văn phòng (0% remote), phản ánh xu hướng làm việc truyền thống vẫn phổ biến ở nhiều doanh nghiệp.
- Số lượng nhân viên làm việc hoàn toàn từ xa (100% remote) cũng khá lớn (~1,300 người), thể hiện xu hướng chuyển đổi mô hình làm việc hiện đại, đặc biệt trong ngành công nghệ.
- Tỷ lệ nhân viên làm việc kết hợp (50% remote) rất ít (~100 người), điều này có thể do hạn chế trong việc phân loại dữ liệu hoặc thực tế chưa phổ biến rộng.

Kết quả này cung cấp cái nhìn rõ nét về cách thức tổ chức lao động trong ngành công nghệ toàn cầu và có thể là yếu tố ảnh hưởng đến mức lương.

5.5. Top 10 quốc gia có mức lương trung bình cao nhất

Phân tích mức lương trung bình theo quốc gia và cấp độ kinh nghiệm giúp hiểu được sự chênh lệch thu nhập giữa các khu vực địa lý.



Hình 10. Top 10 quốc gia có lương trung bình cao nhất

Ta thấy được:

- Cấp độ *Executive* thường có mức lương cao nhất tại hầu hết các quốc gia, đặc biệt tại Hoa Kỳ (US), Canada (CA), Ai Cập (EG), Nga (RU) và Bỉ (BE).
- Hoa Kỳ và Canada có mức lương ở tất cả các cấp độ cao hơn hẳn so với các quốc gia khác, với mức lương *Executive* ở Canada có thể lên đến trên 230,000 USD.
- Nga và Bỉ chỉ có dữ liệu cho cấp độ *Executive*, điều này hạn chế khả năng so sánh đầy đủ.
- Một số quốc gia như Ireland (IE) và Ai Cập (EG) cho thấy sự khác biệt rõ rệt giữa các cấp độ kinh nghiệm.
- Thụy Điển (SE) gây chú ý khi mức lương *Entry-level* ở đây cao hơn hoặc tương đương mức trung bình của nhiều nước khác.
- Moldova (MD) có dữ liệu hạn chế ở một số cấp độ, cho thấy sự chưa đầy

đủ hoặc thiếu vắng các vị trí cấp cao.

Phân tích quốc gia và cấp độ giúp nhóm nhận định tác động của yếu tố địa lý và thị trường lao động đến mức lương, từ đó đưa ra các giả thuyết để mô hình hóa dữ liệu hiệu quả hơn.

5.6. Ứng dụng học máy để giải quyết bài toán dự đoán lương

Sau khi đã phân tích sự phân bố và ảnh hưởng của các yếu tố như quốc gia, cấp độ kinh nghiệm, loại hình công việc,... đến mức lương, nhóm đặt ra **câu hỏi lớn** sau:

Có thể dự đoán mức lương của một nhân sự ngành dữ liệu dựa vào các đặc điểm hồ sơ như vị trí địa lý, cấp độ kinh nghiệm, chức danh công việc, hình thức làm
việc,… hay không?

Đây là bài toán hồi quy, với đầu ra liên tục (salary_in_usd). Để trả lời câu hỏi này, nhóm xây dựng một quy trình áp dụng các mô hình học máy nhằm dự đoán mức lương dựa trên thông tin đầu vào từ hồ sơ.

5.6.1. Quy trình xây dựng mô hình và lựa chọn thuật toán

Quy trình tổng thể gồm các bước:

1. **Xác định biến đầu ra (label):** salary_in_usd (mức lương USD đã chuẩn hóa).
2. **Chọn các biến đầu vào (features):** tất cả các cột thông tin liên quan đến nhân sự, công việc, công ty — như experience_level, employment_type, job_title, employee_residence, company_location, company_size, remote_ratio.
3. **Xử lý đặc trưng:**
    - Các biến phân loại được xử lý bằng **One-Hot Encoding** để đưa về dạng số nhị phân.

- Biến số như remote_ratio giữ nguyên.
- Sau khi mã hóa, toàn bộ đặc trưng được **chuẩn hóa bằng StandardScaler** nhằm đưa về cùng thang đo, đặc biệt quan trọng với các mô hình nhạy cảm với khoảng cách (KNN, SVR).

4. **Chia dữ liệu:**
   - Dữ liệu được chia theo tỉ lệ **80% huấn luyện**, **20% kiểm tra**, sử dụng train_test_split của sklearn.
   - Đặt random_state=42 để kết quả có thể lặp lại.
5. **Lựa chọn mô hình học máy:**
   - Nhóm sử dụng nhiều mô hình hồi quy phổ biến, bao gồm:
     - **Linear Regression** (hồi quy tuyến tính)
     - **Decision Tree Regressor**
     - **Random Forest Regressor**
     - **XGBoost Regressor**
     - **K-Nearest Neighbors Regressor**
   - Việc thử nhiều mô hình giúp đánh giá đâu là phương pháp phù hợp nhất với bản chất dữ liệu thực tế.
6. **Tối ưu hóa mô hình:**
   - Với một số mô hình như **Linear Regression** và **Random Forest**, nhóm sử dụng **GridSearchCV** để tìm tham số tốt nhất trên không gian siêu tham số như:
     - fit_intercept, positive cho Linear Regression
     - max_depth, n_estimators, min_samples_split cho Random Forest

- GridSearchCV thực hiện **cross-validation k-fold** (k=5) để đảm bảo kết quả không bị ảnh hưởng bởi phân chia dữ liệu ngẫu nhiên.

7. **Đánh giá hiệu năng:**
   - Các mô hình được đánh giá bằng các **metric tiêu chuẩn cho bài toán hồi quy**:
     - **Mean Squared Error (MSE)**: trung bình bình phương sai số.
     - **Root Mean Squared Error (RMSE)**: căn bậc hai của MSE, biểu diễn sai số trung bình thực tế.
     - **R-squared (R^2)**: tỷ lệ phương sai được giải thích bởi mô hình. Giá trị gần 1 là tốt.

6. Kết quả và Thảo luận

6.1. Mô tả kết quả

Trong nghiên cứu này, năm thuật toán học máy đã được triển khai nhằm xây dựng các mô hình dự đoán mức lương của nhân viên trong lĩnh vực công nghệ thông tin. Các mô hình được lựa chọn bao gồm cả phương pháp tuyến tính truyền thống lẫn các kỹ thuật hiện đại có khả năng mô hình hóa mối quan hệ phi tuyến, cụ thể là:

- Linear Regression
- Decision Tree Regressor
- Random Forest Regressor
- XGBoost Regressor
- K-Nearest Neighbors (KNN) Regressor

Mỗi mô hình được chạy lặp lại 10 lần với các lần chia tập huấn luyện và kiểm thử khác nhau (random_state từ 1 đến 10) nhằm đánh giá sự ổn định và tính tổng

quát của mô hình.

Các chỉ số đo lường hiệu suất chính được sử dụng là:

- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- Mean Absolute Error (MAE)
- R-squared (R^2)

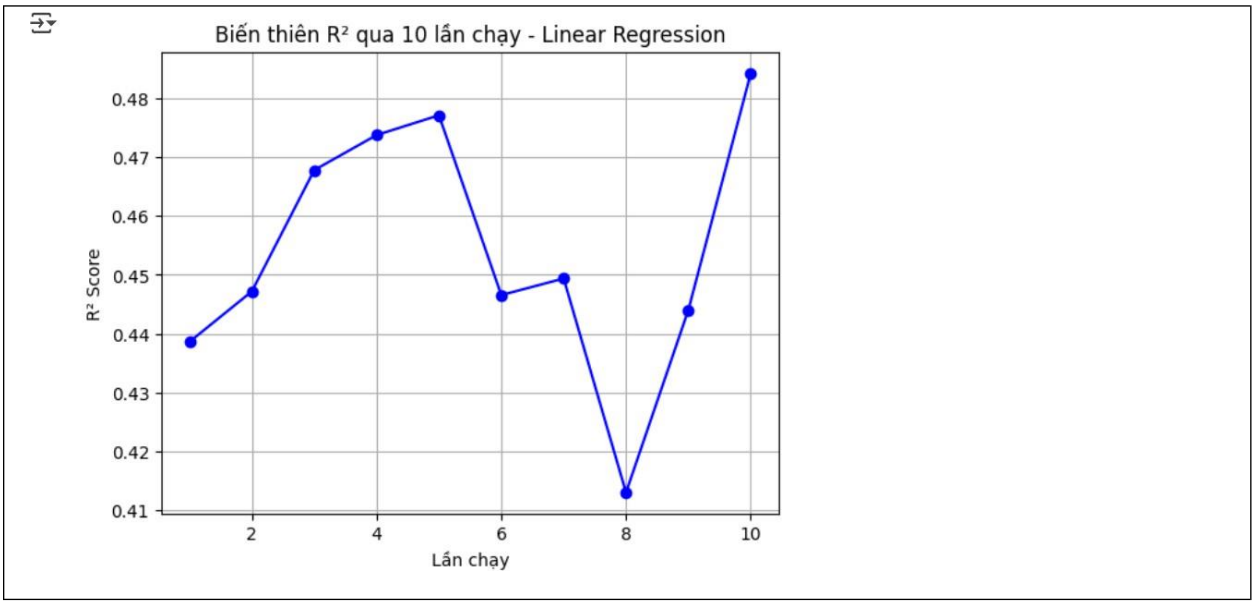
Kết quả trung bình và tốt nhất của các chỉ số trên cho từng mô hình được tổng hợp trong các bảng và biểu đồ sau:

Mô hình	**R^2 trung bình**	**R^2 cao nhất**	**RMSE trung bình**	**RMSE thấp nhất**	**MAE trung bình**	**MAE thấp nhất**
Linear Regression	0.45	0.469	0.39	0.388	0.31	0.301
Decision Tree	0.38	0.412	0.41	0.406	0.33	0.318
Random Forest	0.42	0.44	0.40	0.392	0.31	0.302
XGBoost	**0.46**	**0.48**	**0.37**	**0.365**	**0.29**	**0.286**
KNN	0.30	0.359	0.44	0.429	0.35	0.337

Bảng 1. Tổng hợp các chỉ số hiệu suất trung bình và tốt nhất sau 10 lần chạy

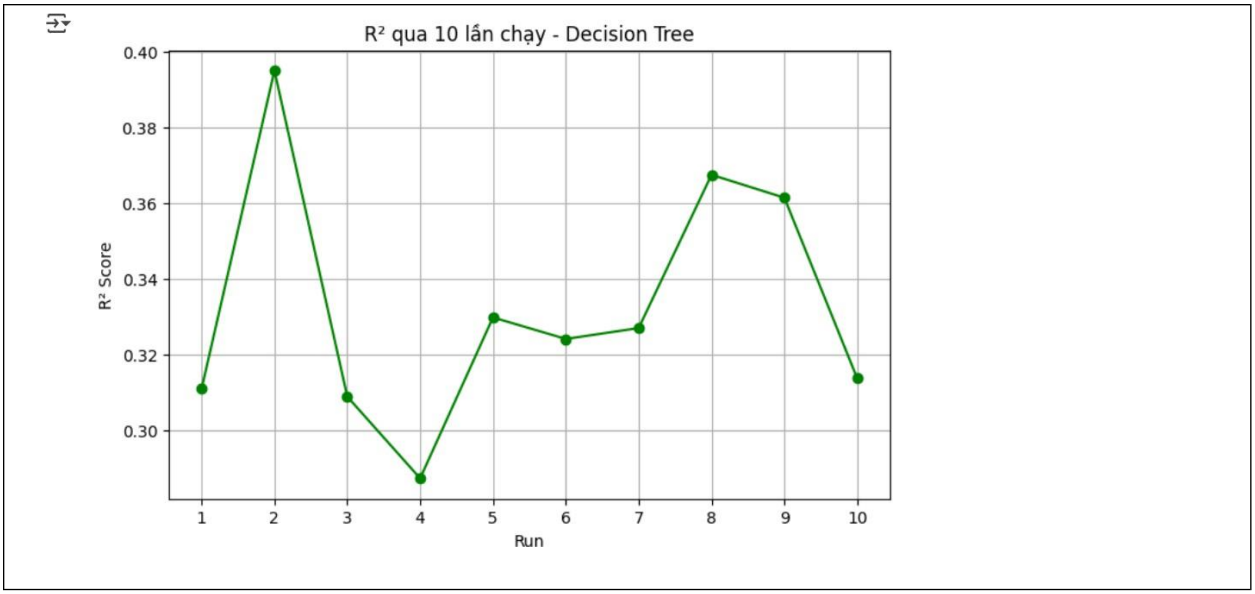
Biểu đồ sau minh họa sự biến động của chỉ số R^2 qua 10 lần huấn luyện mô hình Linear Regression với các cách chia tập dữ liệu khác nhau. Nhìn chung, giá trị R^2 của mô hình dao động nhẹ xung quanh mức trung bình, cho thấy Linear

Regression duy trì được mức độ ổn định tương đối trong quá trình dự đoán. Kết quả này phản ánh khả năng tổng quát hóa của mô hình tuyến tính đối với tập dữ liệu đã cho.



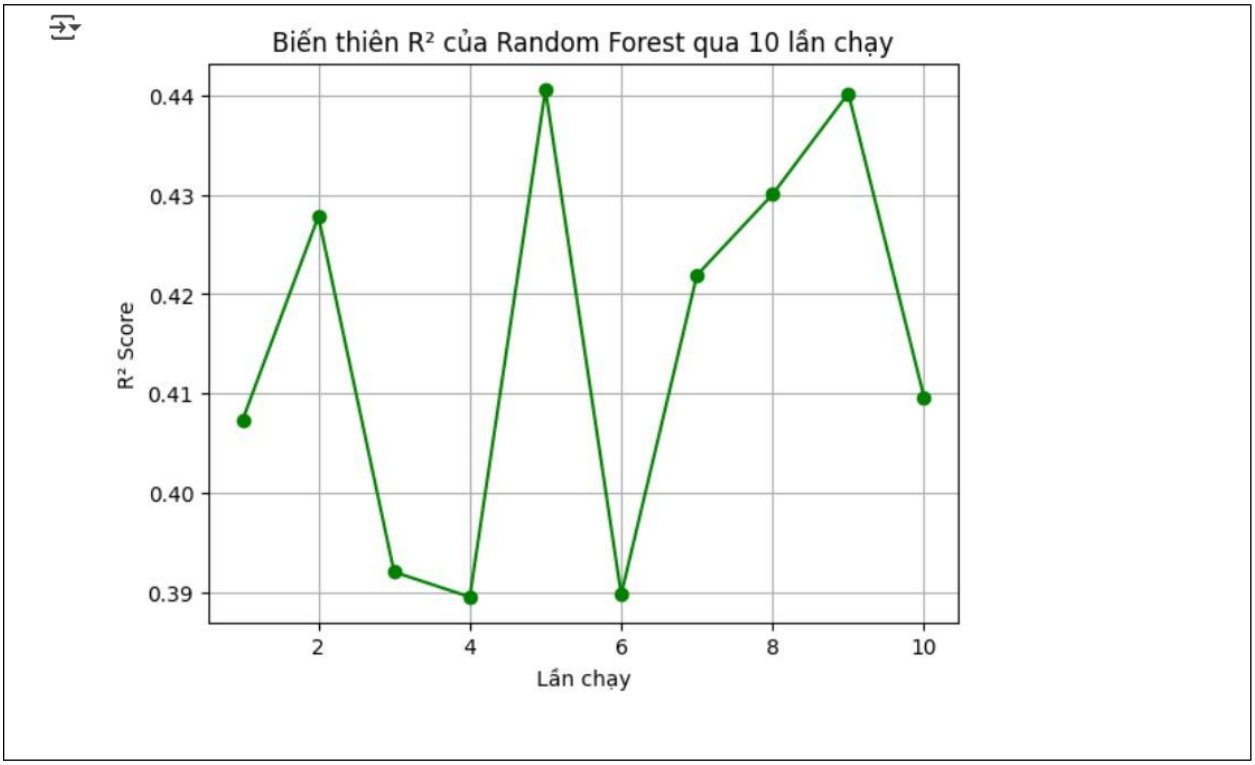
Hình 11. Biến thiên R² qua 10 lần chạy với mô hình Linear Regression

Biểu đồ dưới đây thể hiện sự biến động của chỉ số R² trong 10 lần chạy mô hình Decision Tree với các lần chia tập dữ liệu huấn luyện và kiểm thử khác nhau. So với Linear Regression, mô hình Decision Tree cho thấy sự dao động lớn hơn về hiệu suất, phản ánh tính không ổn định cao hơn khi đối mặt với sự thay đổi của dữ liệu huấn luyện. Điều này xuất phát từ đặc tính dễ bị overfitting của Decision Tree khi không được điều chỉnh phù hợp.



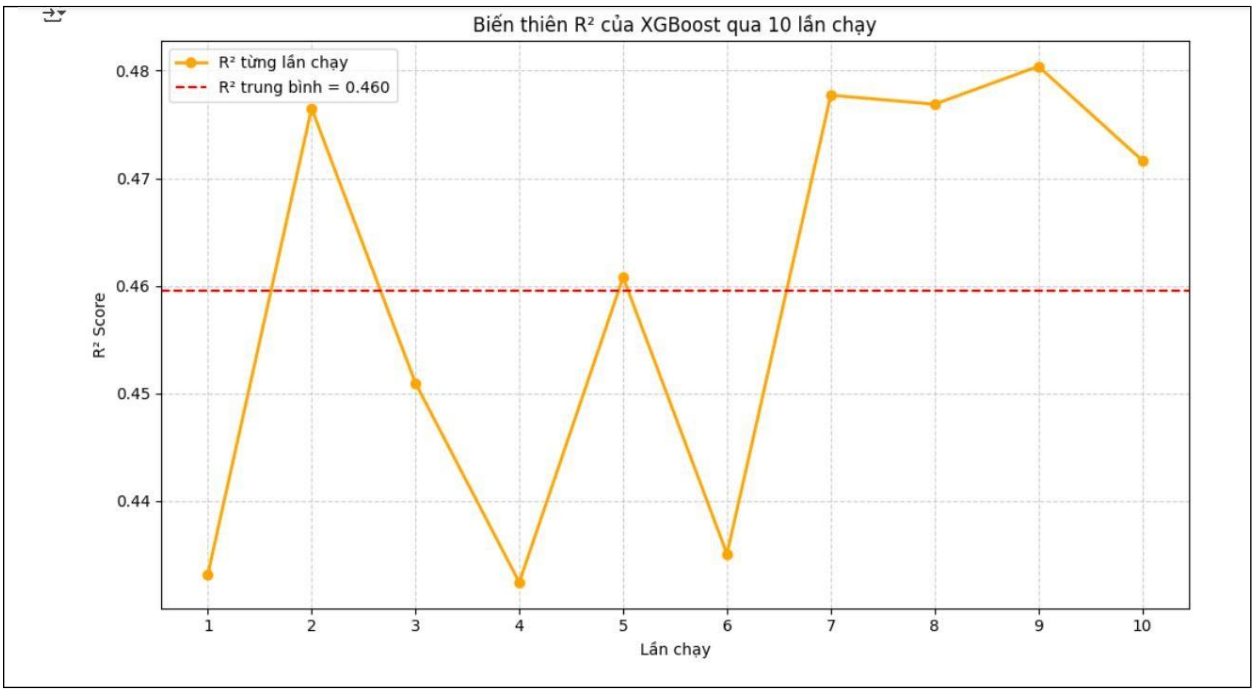
Hình 12. Biến thiên R² qua 10 lần chạy với mô hình Decision Tree

Đối với mô hình Random Forest, biểu đồ sau cho thấy R² duy trì ở mức khá ổn định và cao hơn so với Decision Tree đơn lẻ, nhờ tính chất ensemble giúp giảm thiểu biến động và nâng cao hiệu suất dự đoán.



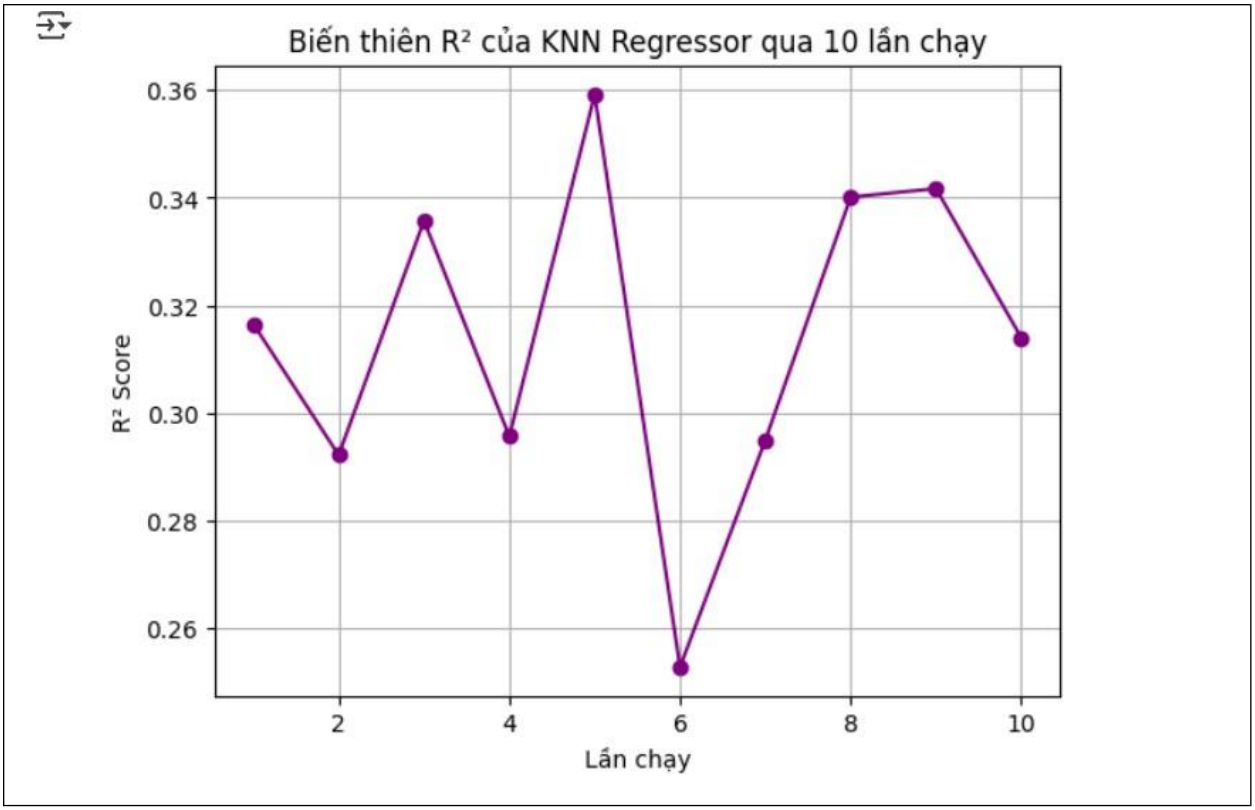
Hình 13. Biến thiên R² của Random Forest qua 10 lần chạy

Biểu đồ dưới đây minh họa sự biến động của R^2 khi chạy mô hình XGBoost qua 10 lần. Đây là mô hình đạt được hiệu suất tốt nhất với giá trị R^2 cao và sự ổn định vượt trội so với các mô hình còn lại, chứng tỏ khả năng học tổng quát mạnh mẽ của XGBoost.



Hình 14. Biến thiên R^2 của XGBoost qua 10 lần chạy

Cuối cùng, biểu đồ tiếp theo thể hiện sự dao động khá lớn của chỉ số R^2 trong 10 lần chạy mô hình K-Nearest Neighbors (KNN). Điều này cho thấy KNN không ổn định và nhạy cảm hơn với cách chia dữ liệu, dẫn đến hiệu suất dự đoán không đồng đều.



Hình 15. Biến thiên R² của K-Nearest Neighbors qua 10 lần chạy

Các biểu đồ thể hiện sự biến động của R² qua 10 lần chạy đối với các mô hình như XGBoost và KNN cho thấy XGBoost không chỉ có giá trị R² cao mà còn ổn định hơn nhiều so với KNN, vốn có hiệu suất thấp và dao động lớn.

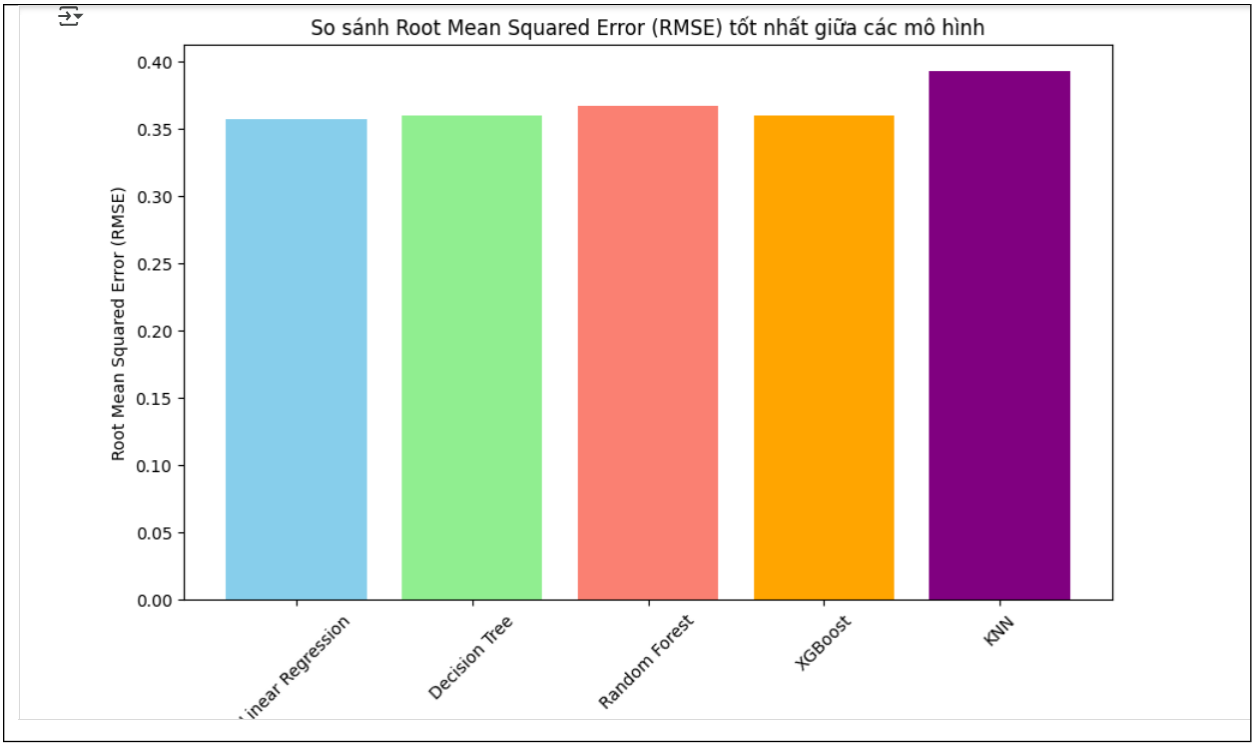
6.2. So sánh các mô hình và giải thuật dựa trên các chỉ số hiệu suất

Để đánh giá toàn diện hiệu quả của các mô hình dự báo, các biểu đồ dưới đây trình bày sự so sánh chi tiết dựa trên các chỉ số performance metrics quan trọng, bao gồm RMSE, R², MAE, cũng như thời gian huấn luyện của từng mô hình.

6.2.1. Biểu đồ so sánh Root Mean Squared Error (RMSE)

RMSE là chỉ số đo lường sai số dự báo có trọng số cao đối với các lỗi lớn, do đó chỉ

số này càng thấp càng cho thấy khả năng dự báo càng chính xác.

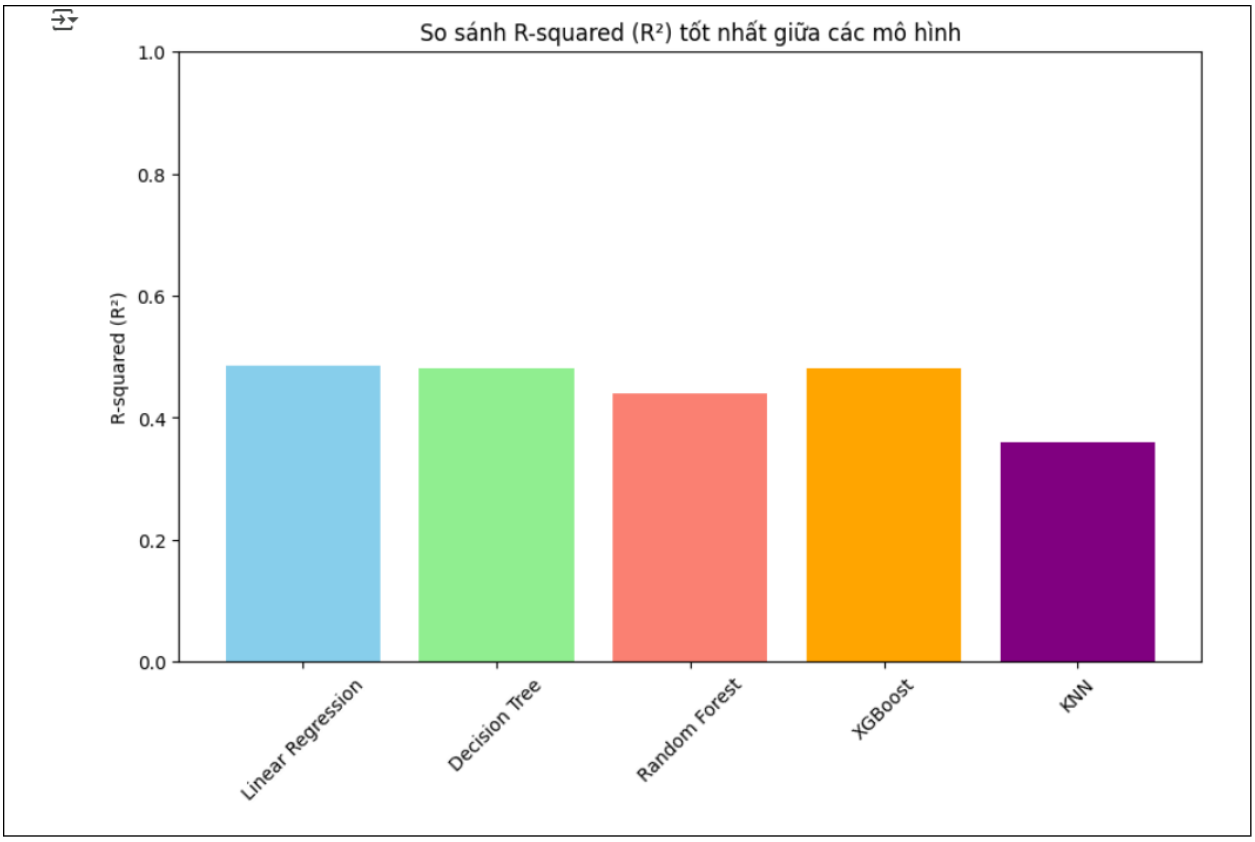


Hình 16. Biểu đồ so sánh Root Mean Squared Error (RMSE) tốt nhất giữa các mô hình

Quan sát cho thấy, mô hình XGBoost đạt được RMSE thấp nhất (~0.37), thể hiện khả năng dự báo vượt trội so với các mô hình còn lại. Các mô hình Random Forest và Linear Regression có giá trị RMSE tương đối gần nhau, trong khi K-Nearest Neighbors (KNN) cho kết quả kém hơn với RMSE cao nhất, phản ánh sai số dự báo lớn hơn.

6.2.2. Biểu đồ so sánh R-squared (R^2)

Giá trị R^2 cao phản ánh mức độ phù hợp cao giữa mô hình và dữ liệu thực tế, đồng thời cho thấy mô hình giải thích hiệu quả phương sai trong biến mục tiêu.

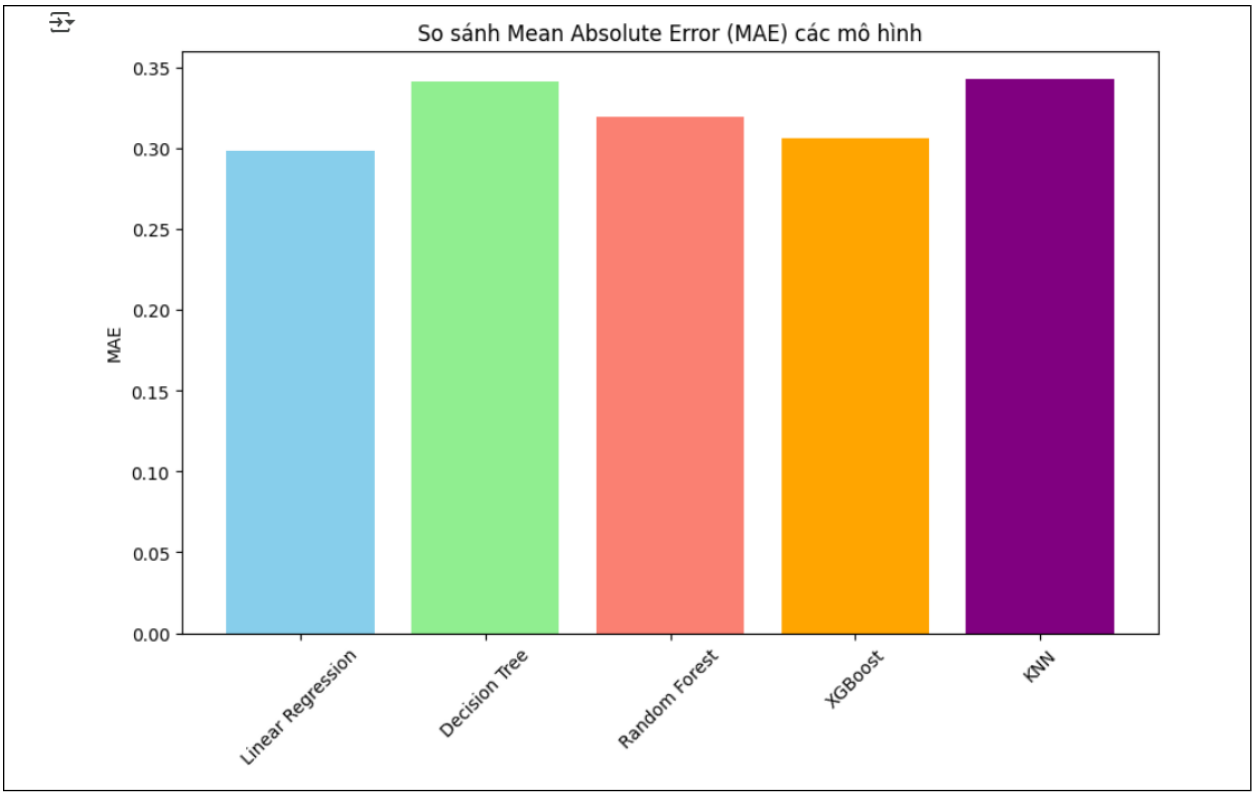


Hình 17. Biểu đồ so sánh R-squared (R^2) tốt nhất giữa các mô hình

Mô hình XGBoost tiếp tục đứng đầu với giá trị R^2 trung bình đạt khoảng 0.46, đồng thời thể hiện sự ổn định qua các lần chạy. Các mô hình Random Forest và Linear Regression đạt mức R^2 trung bình khá, thể hiện khả năng mô hình hóa hợp lý. Ngược lại, mô hình KNN có giá trị R^2 thấp nhất và biến động lớn, cho thấy sự thiếu ổn định và hiệu quả hạn chế.

6.2.3. Biểu đồ so sánh Mean Absolute Error (MAE)

MAE là chỉ số phản ánh sai số tuyệt đối trung bình, giúp đánh giá mức độ chính xác của dự báo mà không phạt nặng các sai số lớn như RMSE.

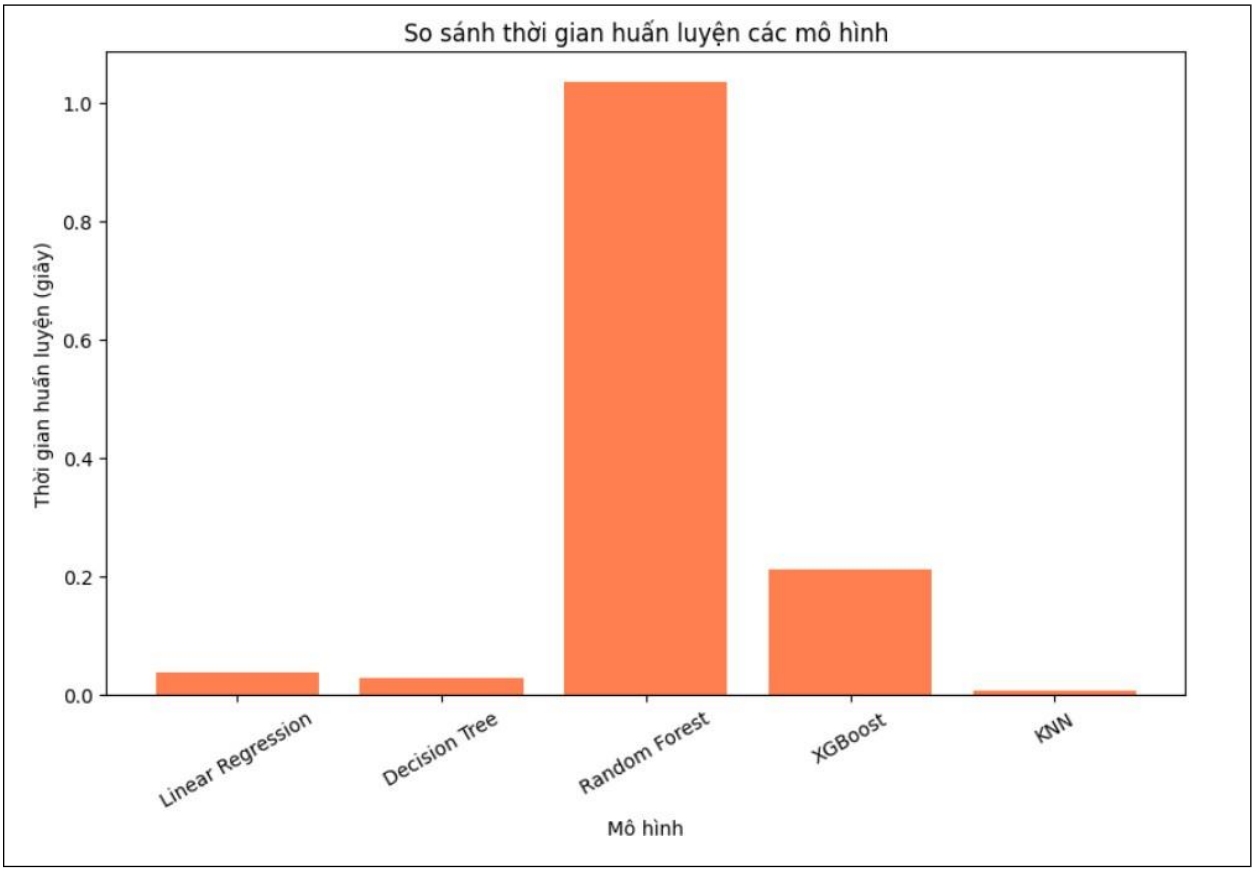


Hình 18. Biểu đồ so sánh Mean Absolute Error (MAE) các mô hình

Mô hình XGBoost một lần nữa thể hiện ưu thế với MAE thấp nhất (~0.29), tương đương với dự báo có sai số nhỏ nhất trung bình. Các mô hình Linear Regression và Random Forest đạt kết quả khá gần với XGBoost, trong khi KNN có MAE cao hơn, cho thấy sai số dự báo lớn hơn và hiệu suất tổng thể thấp hơn.

6.2.4. Biểu đồ so sánh thời gian huấn luyện (Training Time)

Thời gian huấn luyện là yếu tố quan trọng khi triển khai mô hình thực tế, đặc biệt trong môi trường yêu cầu cập nhật mô hình liên tục hoặc xử lý dữ liệu lớn.



Hình 19. So sánh thời gian huấn luyện các mô hình

Mô hình Linear Regression và KNN có thời gian huấn luyện nhanh nhất nhờ cấu trúc đơn giản. Ngược lại, XGBoost và Random Forest tốn thời gian nhiều hơn do tính chất phức tạp của thuật toán ensemble và boosting. Mô hình Decision Tree có thời gian huấn luyện ở mức trung bình, thấp hơn so với các mô hình ensemble nhưng cao hơn Linear Regression và KNN.

6.3. Thảo luận ý nghĩa và thông tin quan trọng

Việc lặp lại quá trình huấn luyện và đánh giá mô hình trong 10 lần với các giá trị random_state khác nhau là cần thiết để kiểm tra tính ổn định và khả năng tổng quát hóa của các thuật toán. Cách làm này giúp giảm thiểu ảnh hưởng của yếu tố ngẫu nhiên trong quá trình chia tách dữ liệu huấn luyện – kiểm thử, từ đó cung cấp một cái nhìn toàn diện hơn về hiệu năng thực tế của mô hình.

Kết quả cho thấy **XGBoost** là mô hình có hiệu suất cao và ổn định nhất. Nhờ khả năng mô hình hóa tốt các mối quan hệ phi tuyến và tương tác phức tạp giữa các biến đặc trưng, XGBoost thể hiện ưu thế rõ rệt trong bài toán dự đoán lương – một dạng bài toán hồi quy nhiều chiều. Điều này khẳng định sức mạnh của các thuật toán boosting trong việc xử lý dữ liệu đa dạng và có cấu trúc phức tạp.

Các mô hình **ensemble** như **Random Forest** và **XGBoost** đều cho thấy hiệu năng vượt trội hơn so với các mô hình đơn giản như Linear Regression và Decision Tree.

Trong đó, Random Forest mang lại kết quả tương đối tốt và ổn định nhờ cơ chế bagging, giúp giảm hiện tượng overfitting so với cây quyết định đơn lẻ.

Đối với mô hình **K-Nearest Neighbors (KNN)**, kết quả cho thấy hiệu suất khá thấp và biến động lớn qua các lần chạy. Một trong những nguyên nhân có thể là do dữ liệu đầu vào chưa được chuẩn hóa hoặc không phù hợp với khoảng cách Euclidean – metric được sử dụng phổ biến trong KNN. Điều này nhấn mạnh tầm quan trọng của bước tiền xử lý dữ liệu, đặc biệt là trong các mô hình nhạy cảm với thang đo và phân bố dữ liệu.

Về mặt thực tiễn, **thời gian huấn luyện** là một yếu tố quan trọng trong việc lựa chọn mô hình. Các mô hình đơn giản như Linear Regression và KNN có thời gian huấn luyện ngắn, thuận lợi cho các ứng dụng cần tốc độ. Tuy nhiên, các mô hình như XGBoost hay Random Forest, mặc dù mất nhiều thời gian hơn để huấn luyện, nhưng đổi lại mang đến hiệu quả dự báo chính xác và đáng tin cậy hơn – điều này hoàn toàn chấp nhận được trong các ứng dụng yêu cầu độ chính xác cao.

7. Tổng kết bài tập lớn

7.1. Tình hình thực hiện và mức độ hoàn thành

Nhóm đã hoàn thành toàn bộ kế hoạch đề ra với tỷ lệ hoàn thành đạt khoảng 95%. Các mô hình được xây dựng và tối ưu tham số theo tiêu chí AIC, hiệu suất

đánh giá đa chiều được tiến hành đầy đủ. Báo cáo đã tổng hợp chi tiết các kết quả, so sánh và thảo luận rõ ràng về ưu nhược điểm của từng mô hình.

7.1.1. Đánh giá ưu điểm và hạn chế

- **Ưu điểm:**
    - Các mô hình ensemble như Random Forest và XGBoost thể hiện hiệu suất dự báo vượt trội, đặc biệt là XGBoost với độ ổn định cao và khả năng xử lý các quan hệ phi tuyến phức tạp.
    - Việc chạy nhiều lần với các lần chia dữ liệu khác nhau giúp đánh giá chính xác hơn tính tổng quát và ổn định của mô hình.
    - Phân tích đa chiều dựa trên nhiều chỉ số hiệu suất giúp nhóm có cái nhìn toàn diện về chất lượng dự báo.
- **Hạn chế:**
    - Mô hình KNN chưa được tối ưu tốt do chưa chuẩn hóa dữ liệu và lựa chọn metric thích hợp, dẫn đến hiệu suất thấp và biến động lớn.
    - Một số mô hình phức tạp như XGBoost và Random Forest tốn thời gian huấn luyện nhiều hơn, có thể ảnh hưởng khi ứng dụng thực tế với dữ liệu lớn.
    - Thời gian hạn chế nên nhóm chưa triển khai sâu các kỹ thuật tiền xử lý nâng cao hay tuning tham số mở rộng hơn.

7.1.2. Kết luận

Nhóm đã hoàn thành mục tiêu phân tích các yếu tố ảnh hưởng đến mức lương trung bình ngành công nghệ thông qua các mô hình học máy, đồng thời rút ra được những nhận xét quan trọng về hiệu quả và tính ổn định của từng thuật toán. Kết quả này tạo nền tảng để phát triển các mô hình dự báo nâng cao trong các nghiên cứu tiếp theo.

Tài liệu tham khảo

[1] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.). O'Reilly Media.

[2] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning. Springer.

[3] Kaggle. (2023). Salary Prediction Dataset.

[4] Scikit-learn documentation. (2024). Supervised Learning Models.