AI Project: Salary Prediction Overview

Step	Algorithm / Model	Purpose
1	Linear Regression	Baseline salary predictor (continuous)
2	Random Forest Regressor	Nonlinear ensemble model capturing complex feature interactions
3	One-Hot Encoding	Convert categorical columns (job title, country) into numeric features
4	Standard Scaler	Normalize numeric features (years, benefits score)
5	Column Transformer + Pipeline	Combine preprocessing & model into one workflow
6	Feature Importance	Ranks most influential variables on salary
7	Evaluation Metrics	R^2, RMSE to measure performance

Libraries & Tools:
- numpy, pandas (data handling)
- matplotlib, seaborn (visualization)
- scikit-learn (LinearRegression, RandomForestRegressor, pipelines, preprocessing, metrics)
- xgboost (installed but not fully used)
- joblib (save/load trained model)

Workflow:
1. Load CSV & inspect dataset
2. Clean & engineer features (year, encode categorical)
3. Explore salary trends (plots by job title, location, experience)
4. Train & evaluate models:
- Linear Regression (R^2 ~0.84, RMSE ~23.8k)
- Random Forest (R^2 ~0.87, RMSE ~21.1k)
5. Interpret feature importance
6. Save model & predict new salaries

Key Takeaways:
- Predicts AI job salaries using regression algorithms.
- Random Forest performs better than Linear Regression.
- Features like years_experience, experience_level, and location strongly influence salary.
- Pipeline ensures reproducible preprocessing and prediction.