**Trevor Zeiger**

**DSC – 680**

**Week 4.2 Porfolio milestone**

**Link**

**https://github.com/Zeigs03/Portfolio.git**

**GitHub Portfolio Projects: Top 10 Showcases**

1. **Remote Work Salary Analysis**  
   *An in-depth analysis of remote job salaries across industries using data mining and machine learning. The project includes data cleansing, EDA, regression modeling, and visualization of salary trends.*
2. **Retail Sales Forecasting**  
   *Time series forecasting using historical retail sales data to predict future trends. Techniques included ARIMA and SARIMA models, with a focus on seasonal demand patterns.*
3. **Movie Rating Predictor (OMDb API)**  
   *A machine learning model that predicts movie ratings using metadata from the OMDb API. This includes data scraping, feature engineering, and regression modeling.*
4. **Multiclass Classification Model Evaluation**  
   *Built and evaluated classification models using weighted and macro averaging techniques. Focused on understanding precision, recall, and F1-score across multiple classes.*
5. **Pandas Series Exploratory Analysis**  
   *Exploration of real-world data using Pandas Series, including descriptive statistics, filtering, and visualization to identify key patterns and outliers.*
6. **Telework Job Postings Data Cleaning**  
   *Cleaned and standardized a complex job posting dataset to prepare for analysis. Included normalization of salary data, job types, and contract fields.*
7. **Data Mining Problem Formulation**  
   *Designed an original data mining problem focused on identifying ideal remote work conditions. Built a classification model to predict job satisfaction based on job descriptions.*
8. **Remote Work Salary Trends**  
   *A analysis project analyzing remote work compensation trends using machine learning and data visualization. Includes policy implications and recommendations.*
9. **Real-Time Retail Fraud Detection Concept**  
   *Conceptualized a fraud detection system using real-time transaction monitoring. Outlined system architecture, key metrics, and anomaly detection techniques.*
10. **Financial Analytics Career Transition Strategy**  
    *Documented a self-guided learning and project roadmap to transition into analytics roles in finance. Focused on SQL, data modeling, and portfolio alignment.*

README 1  
# Remote Work Salary Analysis

## Overview

This project explores trends in remote work compensation across various industries using real-world job posting data. The analysis includes data preprocessing, exploratory data analysis (EDA), regression modeling, and visualizations to uncover salary insights.

## Objectives

- Understand salary ranges and averages for remote positions.

- Identify salary disparities across industries, job types, and experience levels.

- Predict salary based on job features using regression models.

## Features

- \*\*Data Cleaning:\*\* Normalization of job types, contract types, and salary formats.

- \*\*EDA:\*\* Distributions, salary trends, and outlier analysis.

- \*\*Regression Models:\*\* Linear Regression, Decision Tree Regression.

- \*\*Visualizations:\*\* Salary distributions, industry trends, and correlation heatmaps.

## Tools & Technologies

- Python

- Pandas, NumPy

- Scikit-learn

- Matplotlib, Seaborn

## Files

- `data\_cleaning.py`: Scripts to clean and normalize job data.

- `eda.ipynb`: Interactive analysis and visualizations.

- `modeling.py`: Salary prediction models and evaluation.

- `README.md`: Project documentation.

## Future Improvements

- Incorporate NLP techniques to extract features from job descriptions.

- Extend analysis to include hybrid and in-office positions for comparison.

## Author

Trevor Zeiger

README 2

# Movie Rating Predictor (OMDb API)

## Overview

A machine learning project that predicts IMDb ratings based on movie metadata pulled from the OMDb API. This includes data scraping, feature engineering, and regression modeling.

## Objectives

- Gather structured movie data using an external API.

- Predict IMDb ratings using relevant features such as genre, runtime, and release year.

- Evaluate model performance and improve prediction accuracy.

## Features

- \*\*OMDb API Integration:\*\* Fetch metadata for a list of movies.

- \*\*Data Wrangling:\*\* Convert categorical features (e.g., genre) into model-friendly formats.

- \*\*Feature Engineering:\*\* Derive new variables (e.g., decade, genre count).

- \*\*Modeling:\*\* Linear Regression, Random Forest Regressor.

- \*\*Evaluation:\*\* R², MAE, RMSE.

## Tools & Technologies

- Python

- OMDb API

- Pandas, Scikit-learn

- Requests, Matplotlib

## Files

- `fetch\_movies.py`: API calls and data retrieval.

- `prepare\_data.py`: Data transformation and cleaning.

- `model\_training.ipynb`: Model training and evaluation.

- `README.md`: Project overview and usage.

## API Key

This project uses a private OMDb API key for access. Replace the placeholder in `fetch\_movies.py` with your own.

## Author

Trevor Zeiger