DATA SCIENCE

PROJECT PORTFOLIO

A Collection of Data Analysis & Machine Learning Projects

Data Science Internship

Submitted by

Sahir Dev

Data Science Intern

GitHub: github.com/sahirdev

Submitted to

Unified Mentor Private Limited

30th December, 2025

Projects Included:

E-Commerce Furniture Analysis | Netflix Data Analysis
Instagram Influencers | Google Play Store Apps
Data Science Salaries | Life Expectancy | Customer Satisfaction

Table of Contents

Full Portfolio Repository: https://github.com/sahir-dev

PROJECT 1

E-Commerce Furniture Sales Analysis

Dataset: 2,000 AliExpress furniture products

GitHub: ecommerce-furniture-analysis

Introduction

This report presents an analysis of furniture product sales data from AliExpress, one of the largest e-commerce platforms globally. The goal was to understand what factors influence product sales and whether we could build a model to predict sales performance based on available product attributes.

The motivation behind this analysis came from a simple observation: some products sell thousands of units while others sit at zero. What's different about them? Is it the price? The discount? Something in the product title? I wanted to find out.

Dataset Overview

The dataset contains 2,000 furniture product listings scraped from AliExpress. Each record includes product information that's publicly visible on the platform.

Metric	Value
Total Products	2,000
Total Units Sold	46,987
Average Price	$156.78
Median Price	$113.99
Products with Zero Sales	857 (42.9%)
Products with Discounts	487 (24.4%)

Data Cleaning

The raw data required several preprocessing steps before analysis. Prices came with dollar signs and commas (e.g., '$1,299.99') which I stripped out and converted to numeric values. The originalPrice field was missing for 75.7% of products, which actually turned out to be useful - products without an originalPrice don't show a discount to customers. I created a binary 'has_discount' feature from this.

I also engineered several features: has_discount (whether the product shows a strikethrough price), discount_pct (percentage discount), free_shipping (binary), category (extracted from title), title_length, title_word_count, and price_tier (Budget/Mid-Range/Premium/Luxury).

Exploratory Data Analysis

Sales Distribution

The sales distribution is heavily right-skewed. Most products have very few sales, while a small number of products are responsible for the bulk of total volume. The mean is 23.5 units but the median is only 3 - classic long-tail distribution. About 43% of products have zero sales, and the top 10 products account for over 25% of all sales.

Category Breakdown

Category	Products	Total Sales	Avg Sales
Chair	397	16,579	41.8
Table	588	14,821	25.2
Sofa	406	4,026	9.9
Bed	247	3,304	13.4
Desk	167	3,847	23.0
Dresser	95	3,659	38.5
Other	100	751	7.5

Key Findings

The Discount Effect (Main Finding)

This was the most significant finding of the entire analysis. Products with a visible discount sell 9.8 times more on average than products without one.

	Avg Sales	Median Sales	Count
No Discount	7.7	1	1,513
Has Discount	75.4	18	487

Only 24% of products have a discount shown. This suggests a massive opportunity for sellers - simply adding a strikethrough price could dramatically improve sales. The psychological effect of seeing a 'deal' seems to be very powerful on this platform.

Price Sweet Spot

Budget items (under $50) sell about 20x more than luxury items on average. The sweet spot appears to be in the $30-$100 range, balancing volume with reasonable margins. AliExpress shoppers are clearly value-conscious.

Free Shipping

94% of products already offer free shipping, so it's essentially table stakes now. There's no significant sales advantage to offering it, but removing it would likely hurt. It's become expected rather than a differentiator.

Machine Learning Model

I trained several regression models to predict the number of units sold based on product features. The target variable (sales) was log-transformed due to its heavily skewed distribution.

Model	R^2 Score	RMSE	MAE (units)
Random Forest	0.352	1.10	16.5
Ridge Regression	0.328	1.12	16.7
Linear Regression	0.328	1.12	16.7
Gradient Boosting	0.327	1.12	19.7
Lasso	0.287	1.15	16.9

Random Forest performed best with an R^2 of 0.352. Honestly, this isn't great - the model only explains about 35% of the variance in sales. But given that we're missing important factors like product images, seller reputation, reviews, and advertising spend, this is reasonable.

Price is by far the most important predictor (48.1%), followed by discount percentage (20.1%). This aligns with what we saw in the EDA - pricing strategy matters a lot on this platform.

Recommendations

Based on the analysis, here are my recommendations for furniture sellers on AliExpress:

Use Discounts Strategically: With discounts driving nearly 10x more sales, sellers should seriously consider adding strikethrough pricing. Even a modest 20-30% discount display could significantly boost sales.

Price for Volume: On AliExpress, lower prices win. The data strongly suggests positioning products in the $30-$100 range for maximum volume.

Focus on Chairs and Tables: These categories dominate sales. If you're entering the furniture market on AliExpress, start here.

Optimize Product Titles: Title length showed up as a meaningful predictor (13% importance). Longer, more descriptive titles seem to perform better.

Conclusion

This analysis revealed that discounting is the single most powerful lever for driving furniture sales on AliExpress. The 9.8x sales lift from having a visible discount is remarkable, especially considering that only 24% of products currently use this tactic.

Price positioning matters too - budget items significantly outperform premium ones. The machine learning model achieved an R^2 of 0.35, which means there's still a lot of variance unexplained by the features we had access to.

PROJECT 2

Netflix Data: Cleaning, Analysis and Visualization

Dataset: 8,790 Netflix titles (2008-2021)

GitHub: [netflix-data-analysis]

Introduction

Netflix has become one of the biggest streaming platforms in the world, and I was curious to understand their content strategy better. What kind of content do they focus on? Which countries produce the most shows? When do they prefer to release new content?

This project digs into Netflix's catalog of over 8,700 titles to find patterns and insights. I used Python for the analysis, along with libraries like Pandas for data manipulation and Matplotlib/Seaborn for creating visualizations.

Dataset Overview

Attribute	Details
Total Records	8,790
Time Period	2008 - 2021
Number of Features	10
File Format	CSV

The dataset columns include show_id, type (Movie or TV Show), title, director, country, date_added, release_year, rating, duration, and listed_in (genre categories).

Data Cleaning Process

Before diving into the analysis, I needed to clean up the data. Some columns had missing values represented as 'Not Given' instead of actual nulls. I found about 2,588 missing director names and 287 missing country values. Rather than dropping these rows, I kept them and filtered when needed.

The date_added column was stored as text, so I converted it to datetime format. This let me extract useful features like year, month, and day of the week for time-based analysis. The duration column had mixed formats - movies showed minutes while TV shows showed seasons - so I split this into separate columns.

Exploratory Data Analysis

Content Type Distribution

Movies make up about 70% of the catalog (6,126 titles), with TV shows at 30% (2,664 titles). I expected TV shows to have a bigger share given how much Netflix promotes their original series, but movies clearly dominate in terms of quantity.

Geographic Distribution

The United States leads by a huge margin with 3,240 titles (about 37% of all content). India comes second with 1,057 titles - makes sense given Bollywood's massive output. United Kingdom is third with 638 titles.

Rating Analysis

TV-MA (Mature Audiences) is the most common rating with over 3,200 titles. This clearly shows Netflix's focus on adult content. TV-14 comes second. Family-friendly ratings like TV-Y and TV-G make up a much smaller portion.

Growth Over Time

Netflix's content additions show explosive growth starting around 2015-2016. The peak was in 2019 with over 2,000 titles added that year. There's a slight dip in 2020-2021, which might be related to COVID affecting production schedules.

Release Day Patterns

Here's something cool - Friday is clearly Netflix's preferred release day with nearly 2,500 titles dropped on Fridays. This makes total sense from a business perspective. They want people to have fresh content for weekend binge-watching. Weekends actually have the lowest release numbers.

Key Findings

#	Finding	Details
1	Movies dominate	70% of content are movies, 30% TV shows
2	US leads production	37% of all content comes from United States
3	Adult-focused content	TV-MA is the most common rating by far
4	Friday releases	Most content drops on Fridays for weekend viewing
5	Peak in 2019	Content additions peaked before slight decline
6	Short TV runs	Most TV shows have only 1 season
7	Standard movie length	Average movie is ~100 minutes

Conclusion

This project gave me some really interesting insights into how Netflix builds their content library. Despite being known for binge-worthy TV series, movies actually make up the bulk of their catalog.

The Friday release strategy is very deliberate. They clearly understand their audience's viewing habits and time their releases to maximize engagement over weekends.

The heavy focus on TV-MA content shows Netflix isn't trying to be everything to everyone. They've carved out a niche in adult-oriented content, though they do have kids' content as well.

PROJECT 3

Instagram Influencers Analysis

Dataset: 200 Top Instagram Influencers

GitHub: [instagram-influencers-analysis]

Introduction

I worked on this project to understand how Instagram influencers perform in terms of engagement and what factors affect their influence scores. The dataset has 200 top influencers with details like follower count, likes, engagement rates, and country of origin.

My main goals were to clean the messy data, explore patterns through visualizations, and build some ML models to see if we can predict engagement levels. I used Python with pandas, matplotlib, seaborn, and scikit-learn for this analysis.

Data Cleaning

First thing I did was convert all those string values to actual numbers. For example, "475.8m" becomes 475,800,000. I replaced 'b' with billion, 'm' with million, 'k' with thousand, and removed the '%' signs.

For missing countries (62 entries), I just filled them as "Unknown" since we can't really guess where someone is from. The one missing engagement rate was filled with the median value which was 0.88%.

The Engagement Paradox

This was probably the biggest insight from the whole analysis. You'd think more followers means more engagement, but it's actually the opposite! I grouped influencers by follower tiers:

Follower Tier	Avg Engagement Rate
Under 50M	2.01%
50M - 100M	2.22%
100M - 200M	0.97%
Over 200M	0.85%

So basically, smaller accounts have more engaged audiences. This makes sense when you think about it - mega celebrities have lots of passive followers who don't interact much, while mid-tier influencers often have more dedicated fanbases.

Feature Engineering

To get more insights, I created some additional features: likes_per_follower, likes_per_post, avg_likes_pct, log_followers, and eng_category (Low/Medium/High). Out of 200 influencers, 109 fall in Low engagement (under 1%), 61 in Medium (1-3%), and only 30 in High (over 3%).

Machine Learning

I tried two types of models - regression to predict influence scores and classification to predict engagement categories.

Honestly, the regression didn't work great. Linear regression gave R^2 of 0.046 and Random Forest was even worse at -0.004. This tells me that influence score probably depends on factors not in this dataset.

Classification worked better. Random Forest classifier achieved 75% accuracy in predicting Low/Medium/High engagement. The model was best at identifying Low engagement accounts (89% precision) but struggled a bit with Medium and High categories since there are fewer examples.

Key Takeaways

The main takeaway is that follower count isn't everything. Brands looking for influencer partnerships might get better ROI from mid-tier influencers who have more engaged audiences rather than going for the biggest names.

The US market is pretty saturated with influencers, so there might be opportunities in emerging markets like Brazil, India, and Indonesia where there's growing social media presence.

PROJECT 4

Google Play Store Apps Analysis

Dataset: 10,357 Apps + 37,427 User Reviews

GitHub: [google-playstore-analysis]

Introduction

The mobile app market has grown exponentially over the past decade, with Google Play Store being one of the largest platforms for Android applications. Understanding what makes an app successful is crucial for developers and businesses looking to enter or expand in this market.

This project analyzes the Google Play Store dataset to uncover patterns and insights that can help developers make data-driven decisions. I looked at over 10,000 apps and 37,000 user reviews to understand the relationships between ratings, downloads, pricing, and user sentiment.

Data Cleaning

Raw data usually has issues that need to be fixed before analysis. Found one row where the data was shifted - values were in wrong columns. This was causing issues with data type conversions, so I removed it.

I identified 483 duplicate rows which were removed. Several columns needed conversion: Reviews (string to integer), Size (extracted numeric value from 'M' and 'k' suffixes), Installs (removed '+' and ',' characters), Price (removed '$' symbol). The Rating column had 1,474 missing values which I filled using median imputation by category.

Exploratory Analysis

Rating Distribution

App ratings range from 1 to 5, with most apps clustered between 4.0 and 4.5. The mean rating is 4.20 and median is 4.30, indicating a left-skewed distribution. This makes sense - poorly rated apps tend to get removed or updated quickly.

Category Analysis

The FAMILY category has the most apps (1,943), followed by GAME (1,121) and TOOLS (843). However, when looking at total installs, GAME dominates with 31.5 billion downloads - more than any other category.

Free vs Paid Apps

The vast majority of apps (92.6%) are free. Paid apps make up only 7.4% of the market but show some interesting differences:

Metric	Free Apps	Paid Apps
Count	9,592	765
Percentage	92.6%	7.4%
Avg Rating	4.20	4.27
Median Installs	100,000	1,000

Correlation Analysis

The strong correlation between reviews and installs (r=0.63) is the most important finding. It suggests that apps with more user engagement tend to get more downloads - probably because reviews help with visibility in the store.

Sentiment Analysis

The reviews dataset includes sentiment labels and polarity scores. After cleaning, I analyzed 37,427 reviews. About two-thirds of reviews are positive (64.1%), negative at 22.1%, and neutral at 13.8%. The average sentiment polarity is 0.18, confirming a slight positive lean.

Statistical Testing

I ran a t-test to check if paid apps really have higher ratings. The p-value was less than 0.0001, so the difference is statistically significant - paid apps do have higher ratings on average.

The correlation of 0.63 between reviews and installs is also statistically significant ($p < 0.001$). This confirms that the relationship is real and not due to chance.

Key Findings

Market Dominance: FAMILY category has the most apps, but GAME dominates in downloads with 31.5 billion installs.

Free Model Wins: 92.6% of apps are free. While paid apps rate higher, free apps get 100x more downloads.

Engagement Matters: Strong correlation (r=0.63) between reviews and installs. More engagement leads to more visibility.

Underserved Niches: EVENTS and EDUCATION categories have highest ratings but fewer apps - potential opportunity.

Recommendations

Aim for 4.0+ Rating: Most successful apps are rated between 4.0 and 4.5. Focus on user experience and fix bugs quickly.

Consider Freemium Model: Free apps dominate downloads. Consider offering a free version with premium features.

Encourage Reviews: Reviews correlate strongly with installs. Prompt users to leave reviews at the right moment.

PROJECT 5

Data Science Job Salaries Analysis

Dataset: 565 Job Records (2020-2022)

GitHub: data-science-salaries

Introduction

The data science field has seen tremendous growth over the past few years, with increasing demand for skilled professionals across industries. Understanding salary trends is valuable for both job seekers looking to negotiate compensation and employers aiming to offer competitive packages.

This project analyzes a dataset of 607 data science job records collected between 2020 and 2022. The goal is to identify patterns in compensation based on experience level, work arrangement, company characteristics, and geographic location.

Data Preprocessing

I removed 42 duplicate records from the dataset, dropped unnecessary index columns, converted abbreviated codes to readable labels, and transformed remote_ratio into categorical work_type. After cleaning, the final dataset contained 565 unique job records.

Exploratory Analysis

Salary Distribution

The salary distribution shows a right-skewed pattern, with most data scientists earning between $50,000 and $150,000 annually. The mean salary is $110,610 while the median is $100,000.

Experience Level Analysis

Experience level has a strong correlation with salary. Executive-level professionals earn nearly three times more than entry-level employees.

Level	Avg Salary	Median	Count	% of Total
Entry	$61,643	$56,500	88	15.6%
Mid	$87,793	$76,940	208	36.8%
Senior	$138,375	$135,000	243	43.0%
Executive	$199,392	$171,438	26	4.6%

Work Type Analysis

Remote work dominates the data science market, representing over 61% of all positions. Remote jobs also pay better on average:

Work Type	Avg Salary	Count	Percentage
Remote	$120,763	346	61.2%
Onsite	$105,785	121	21.4%
Hybrid	$80,722	98	17.3%

Key Findings

Experience is the strongest salary predictor: Moving from entry to executive level can triple your salary.

Remote work pays more: Remote positions average $120,763 vs $105,785 for onsite - about 14% higher.

The market is growing rapidly: Job postings increased nearly 4x from 2020 to 2022, and salaries rose 28.5%.

Company size matters less than expected: Large and medium companies pay similarly. Small companies pay around 34% less.

Machine Learning Models

Two regression models were trained: Linear Regression and Random Forest. The dataset was split 80/20 for training and testing.

Model	R^2 Score	MAE	RMSE
Linear Regression	0.319	$37,216	$57,210
Random Forest	0.282	$33,513	$58,711

Random Forest achieved lower MAE ($33,513). Top predictors: company location (33%), job title (30%), experience level (22%).

Recommendations

For Job Seekers: Prioritize gaining experience - it's the clearest path to higher pay. Consider remote positions which often pay better. Target US-based companies for highest salary potential.

For Employers: Offering remote options helps attract talent. Expect to pay more for senior and executive talent. Salaries are rising - budget accordingly.

PROJECT 6

Life Expectancy Analysis

Dataset: WHO Data - 193 Countries (2000-2015)

GitHub: life-expectancy-analysis

Introduction

This project focuses on analyzing life expectancy data from the World Health Organization to understand what factors influence how long people live in different countries. The goal was to build a machine learning model that can predict life expectancy based on various health, economic, and social indicators.

Life expectancy is an important measure of a country's overall health and development. By identifying the key factors that affect lifespan, governments and health organizations can make better decisions about where to invest resources.

Dataset Overview

The dataset was obtained from the WHO Global Health Observatory and contains 2,938 records with 22 different variables. These include health metrics like mortality rates, immunization coverage, and disease prevalence, as well as economic factors like GDP and healthcare spending.

Data Cleaning

The dataset had missing values in several columns. Population data had the most gaps (22%), followed by Hepatitis B immunization (19%) and GDP (15%). Missing values were filled using the mean of each respective column.

Outliers were identified using the IQR method. Values falling below Q1 - 1.5×IQR or above Q3 + 1.5×IQR were replaced with the column mean. This affected about 4,100 data points.

Feature Engineering

Three new features were created: Mortality Ratio (adult mortality divided by infant deaths), Immunization Average (mean of Polio, Diphtheria, and Hepatitis B coverage), and GDP × Income (product of GDP and income composition of resources).

Exploratory Analysis

The average life expectancy across all countries and years was about 69 years, with values ranging from 36 to 89 years. This wide range shows the significant disparity in health outcomes between different nations.

Out of all records, about 83% came from developing countries and 17% from developed countries. The average life expectancy differed significantly - 79 years for developed countries versus 67 years for developing ones.

Correlation Analysis

Feature	Correlation	Interpretation
Schooling	+0.75	Strong positive - more education = longer life
Income composition	+0.72	Strong positive - better resources = longer life
BMI	+0.57	Moderate positive - nutrition matters
Adult Mortality	-0.70	Strong negative - higher mortality = shorter life
HIV/AIDS	-0.56	Strong negative - disease burden reduces lifespan

Model Building

Five different regression algorithms were tested: Linear Regression, Ridge Regression, Random Forest, Extra Trees, and Gradient Boosting.

Model	R^2 Score	RMSE (years)	MAE (years)
Extra Trees	0.9739	1.50	0.91
Random Forest	0.9658	1.72	1.08
Gradient Boosting	0.9446	2.19	1.61
Ridge Regression	0.8533	3.56	2.68
Linear Regression	0.8533	3.57	2.68

The Extra Trees Regressor achieved the best performance with an R^2 score of 0.9739, meaning it explains about 97.4% of the variance in life expectancy. The RMSE of 1.50 years indicates that on average, predictions are within 1.5 years of the actual value.

Feature Importance

HIV/AIDS emerged as the dominant predictor, accounting for over half of the model's predictive power (56.4%). This makes sense given that the HIV epidemic had a devastating impact on life expectancy in many African countries during this period. Income composition (11.7%) and Adult Mortality (10.8%) were also significant.

Key Findings

HIV/AIDS Impact: The epidemic is the single most important factor affecting life expectancy globally. Countries with high HIV prevalence showed significantly lower life expectancies.

Education Matters: Years of schooling showed a strong positive correlation (0.75) with life expectancy. Investing in education appears to be an effective way to improve population health.

Development Gap: There's a persistent 12-year gap in life expectancy between developed and developing countries. While both groups improved, this gap didn't narrow significantly.

Positive Global Trend: Global life expectancy improved by about 4 years from 2000 to 2015, reflecting advances in healthcare, nutrition, and disease prevention.

PROJECT 7

Customer Satisfaction Prediction

Dataset: 8,469 Customer Support Tickets

GitHub: customer-satisfaction-prediction

Executive Summary

This project develops a machine learning solution to predict customer satisfaction ratings from support ticket data. By analyzing 8,469 customer support tickets across 42 different tech products, we built predictive models to classify satisfaction levels on a 1-5 scale.

Key achievements: Processed and cleaned 8,469 support tickets with 17 features, engineered 9 new features from existing data, created 11 comprehensive visualizations, trained and compared 5 different ML models, achieved 22.38% accuracy on 5-class classification (vs 20% random baseline).

Introduction

Customer satisfaction is a critical metric for business success, directly impacting customer retention, brand reputation, and revenue growth. In the tech industry, where products are complex and support interactions frequent, understanding what drives satisfaction is essential.

The problem statement: Can we predict customer satisfaction ratings based on support ticket characteristics? This would enable companies to proactively identify at-risk customers and prioritize support resources effectively.

Dataset Overview

Metric	Value
Total Records	8,469
Records with Ratings	2,769
Number of Features	17
Target Variable	Satisfaction Rating (1-5)
Time Period	2020-2023

Methodology

Data Preprocessing

Filtered records with satisfaction ratings (2,769 out of 8,469), removed duplicate ticket IDs, converted date columns to datetime format, handled missing values using median imputation, and encoded categorical variables using Label Encoding.

Feature Engineering

Created 9 new features from existing data: Time-based features (purchase year, month, day of week, quarter), Resolution Hours (time difference between first response and resolution), Age Group (categorical binning of customer age), and Description Length (character count of ticket description).

Exploratory Analysis

The satisfaction ratings are relatively evenly distributed across all 5 classes, with Rating 3 being slightly more common. This balanced distribution is favorable for classification modeling.

Key insights: Average rating is 2.99/5 indicating neutral to slightly negative overall satisfaction. Refund requests and Technical issues generate the most tickets. Support channels are evenly split across Email (26%), Phone (25%), Social Media (24.7%), and Chat (24.3%). Average resolution time is 7 hours.

Model Results

Model	Accuracy	F1 Score	CV Score
K-Nearest Neighbors	22.38%	21.79%	20.05%
Random Forest	21.48%	21.42%	18.96%
Logistic Regression	19.68%	18.95%	20.32%
Decision Tree	18.95%	18.87%	19.73%
Gradient Boosting	18.05%	18.02%	18.37%

While these accuracies may appear modest, with 5 classes, random guessing would yield 20% accuracy. The even distribution of classes makes this a challenging problem, and synthetic elements in ticket descriptions may limit predictive power.

Feature Importance

Random Forest feature importance analysis revealed the most influential predictors: Description_Length (14.2%), Customer Age (12.8%), Product Purchased (11.9%), Ticket Subject (10.4%), and Resolution_Hours (9.8%).

Recommendations

Prioritize Complex Issues: Tickets with longer descriptions should receive priority attention.

Age-Based Strategies: Develop targeted support approaches for different age demographics.

Product-Specific Training: Focus support team training on high-ticket-volume products.

Resolution Time Focus: Implement SLAs to reduce ticket resolution time.

Future Work

Potential areas for improvement include implementing NLP analysis on ticket descriptions for sentiment extraction, applying deep learning models (LSTM, Transformers) for text-based prediction, developing a real-time prediction API for production deployment, and creating an interactive dashboard for business stakeholders.

End of Portfolio

Submitted by

Sahir Dev

Data Science Intern

Submitted to

Unified Mentor Private Limited

30th December, 2025

GitHub Repository Links

Main Profile: https://github.com/sahir-dev

Project 1: E-Commerce Furniture Analysis

Project 2: Netflix Data Analysis

Project 3: Instagram Influencers Analysis

Project 4: Google Play Store Analysis

Project 5: Data Science Salaries

Project 6: Life Expectancy Analysis

Project 7: Customer Satisfaction Prediction

Tools Used: Python, Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn