Project report:

I used a dataset that gives us a lot of data about used cars in egypt. The dataset gives you multiple information about used cars including price, kilometers driven, engine size and what year model it is. Other information is present in the dataset but isn't relevant for my prevailing question of what affects prices the most between kilometer driven, engine size and year it was made in. To figure this out, I wrote a piece of code in Rust that first loads the CSV file and cleans the data by skipping any rows that don't have the necessary information. Once we have a nice, clean dataset, the code calculates some basic statistical measures like mean and standard deviation for kilometers, year, engine, and price, just to help us understand what typical values look like and how spread out our data is. Then it proceeds to compute the correlation between each of these three factors and price, so we can see which ones have a stronger relationship with how much the car costs. Afterward, it fits a linear regression model using a method known as the normal equation to predict price based on the three factors, and it gives us an equation that shows how price might change if we alter the kilometers, the year, or the engine value. Finally, it prints out a mean squared error (MSE), which is a number that shows us how well or badly our regression model is doing at predicting the price—lower MSE means the predictions are closer to reality, and higher MSE means we have some serious gaps in our model. The code I wrote first opens and reads a CSV file containing these car listings. Inside the code, there is a function called load_csv that checks if the file exists and then iterates over each line, skipping the header and invalid rows, and extracting only the numeric values for Year, Engine, Kilometers, and Price. If any of these fields are missing or non-numeric, that row is skipped, ensuring we end up with only clean and consistent data. After that, the code maps these values into tuples like (Kilometers, Year, Engine, Price). Once the data is loaded, the code computes a bunch of helpful statistics like mean and standard deviation for each attribute, so we can understand our dataset's distribution, and then calculates correlations between Price and each of the other factors to see which ones have the strongest linear relationship with cost. Next, the code constructs what's known as a "design matrix" and a target vector from the data so we can fit a multiple linear regression model using the normal equation. This gives us a formula representing Price as a combination of Kilometers, Year, and Engine. Now, looking deeper into the code: it defines helper functions like is_num to check if strings are numeric and rl to read lines from the file. The statistics and correlation computations are done by simple iterative functions that sum up values and apply mathematical formulas for averages, standard deviations, and Pearson correlation. The linear regression part is more interesting: after constructing matrices using ndarray, the code transposes and multiplies them to get (X^T X) and (X^T y). It then uses a Gaussian elimination function gauss to solve the system of equations (X^T X)*b = X^T y, giving us a vector b of coefficients. These coefficients include the intercept and slopes for Kilometers, Year, and Engine. The code then prints out a neat model equation like Price = intercept + slope_km*Kilometers + slope_y*Year + slope_e*Engine. The results showed that year model had the biggest impact on the price of the used car. The correlation value, r, displayed that as the year increases, the car is younger, the price tends to increase. The regression model coefficient for Year was also large and positive. I conclude a slope of approximately 16167, which means that for each year the car is younger, the price increases by the slope's number. This represents another piece of evidence that the newer the car, the higher the prices. In contrast, kilometers driven had a negative correlation with price. The higher the kilometers driven/ mileage was, the cheaper the cars were. Although the correlation between kilometers driven and price is high relative to its negativity, it still does not have as big of an impact as the year does. In comparison to r between year and price= 0.45, the r of kilometers driven and price= -0.42. It also has a slope equal to -1.3421 in the regression equation which indicates that with each kilometer driven the price decreases by -1.3421.Usually kilometers are calculated in bigger ranges and that's why it's much lower than the slopes of the year and the engine. Lastly, the engine did show a positive relationship with price with r= 0.28 but despite that, it's not strong as either of the other two factors. The regression model coefficient for the engine was positive. I conclude a slope of approximately 296, which means that for each additional unit increase in engine size, the price increases by the slope's number.