***LINEAR REGRESSION SALARY***

1. In simple linear regression, Salary=B0+B1\*Years of experience where B1 acts like the slope or gradient which represents how much predicted salary changes with every change in years of experience. A +ve coefficient means salary is going to increase as experience increases the extent of which is represented by the value of the coefficient. The higher it is the more exponential effect it is going to have on salary.

2. If salary or experience have outliers or anomalies then a simple linear regression would be influenced by those values.

3. If you do not reserve any data for testing (or use cross-validation), you cannot reliably estimate how well the model generalizes to unseen data. You may end up overfitting, meaning the model appears to perform very well on the training set but might perform poorly on new data. Your reported performance metrics (e.g., R2R^2, MAE, RMSE) would be overlyoptimistic and not reflect true predictive performance.

4. This indicates a systematic bias in your model. Consistently lower predictions suggest the model is underestimating salaries, possibly because: The intercept (β0\beta\_0) is too low. The slope (β1\beta\_1) does not capture the true rate of increase in salary per year. A relevant predictor is missing (e.g., certifications, education level, or industry). Consistently higher predictions are the opposite: the model is overestimating actual salaries. Checking residual plots (which you have) helps diagnose whether the errors are random or show a pattern indicating bias.

5. Increasing the training set leads to better generalisation and more stable parameter estimates with less variance in coefficients. More data also reduces overfitting. Decreasing the training set increases the risk of overfitting and can result in unstable and high variance coefficient estimates.

6. If certifications explain variations in salary, then it could improve R^2 and improve predictions. The coefficient for experience may change depending on how correlated the certifications with experience are.

7. In a linear regression a single outlier can disturb the slope of the intercept.

8. Model can easily overfit the peculiarities of the training set.

9. In **scikit-learn** (Python), LinearRegression().fit(X, y) implements **Ordinary Least Squares**:

* 1. It reads the feature matrix XX (in this case, years of experience) and the target vector yy (salary).
  2. It solves for the coefficients (β0,β1,…\beta\_0, \beta\_1, \dots) by **minimizing the sum of squared residuals**: min⁡β∑i=1n(yi−y^i)2. \min\_{\beta} \sum\_{i=1}^n (y\_i - \hat{y}\_i)^2.
  3. Returns the fitted model, which includes the **intercept** and **coefficients**.
* Under the hood, it typically uses a matrix solution (XTX)−1XTy(X^TX)^{-1}X^Ty or a similar approach (e.g., Singular Value Decomposition) for numerical stability.

10. The model may shift to accommodate that point worsening the predictions for rest of the data.

11. Current data is for experience 1-20 and 0 does not lie in this range. Observing for data beyond range would be unreliable as model may not hold itself for data like this.

***MLR STARTUP SALARY***

1. Because R&D has the largest coefficient, it appears to have the strongest impact on profit per dollar spent in this dataset. Administration and Marketing have much smaller coefficients. This does **not** mean they are unimportant; rather, it suggests that in this specific model (and dataset), R&D explains most of the variation in profit. The intercept (β0≈15,530\beta\_0 \approx 15{,}530) represents the baseline profit when all spends are zero (which may or may not be realistic).

2. **Mean (70,543.76)** On average, companies in this dataset spend around $70k on R&D. **Standard Deviation (43,042.35)** This is relatively large compared to the mean, indicating **substantial variation** in R&D expenditures.

3. **High Variability:** Marketing Spend ranges from **-185,350** to **682,176**, a difference of nearly $867k. Large ranges can lead to: **Increased sensitivity** to extreme values or outliers. A wide spread in predicted profits for different marketing levels.

4. **Encoding Categorical Variables:** Typically, you would use **one-hot encoding** (dummy variables) for non-ordinal categories, creating new binary columns for each industry type. If the categories have a natural order, you might use **ordinal encoding**, though that’s less common for industry types. **Incorporating Into Regression**: Once encoded, each industry dummy variable would get its own coefficient in the model, capturing differences in profit across industries.

5. **R&D Spend** has a correlation of **0.87** with Profit (very strong positive relationship). **Marketing Spend** has a correlation of **0.37** with Profit (moderate). **Administration** has a near-zero correlation (0.0056) with Profit.

6. **Reason for Splitting:** We split the data into **training** and **test** sets to evaluate how well the model generalizes to **unseen data**. Training on all data can lead to **overfitting**, where the model memorizes the training set but performs poorly on new data. **Default Ratio** In **scikit-learn**, if you do not specify test\_size or train\_size, the default is typically **75% training** and **25% test** (test\_size=0.25).

7. **Little Change in R2R^2** or the model’s predictive power, because Administration does not explain much additional variance. **Slight Simplification** of the model. Removing Administration reduces the number of parameters by one. **Check for Multicollinearity**. If Administration is correlated with R&D or Marketing (though it doesn’t appear so here), removing it could unexpectedly change other coefficients. In this dataset, correlation with R&D or Marketing is quite low, so it likely won’t have a large effect.

8. **Overfitting:** The model may fit the noise or peculiarities in the training data too closely, failing to generalize. **Insufficient or Unrepresentative Data** If the training set does not represent the variety of real-world scenarios, the model may struggle on the test set. **High Variance Model:** If too many variables or complex transformations are used, the model can become unstable when tested on new data.

9. **Potential Extrapolation** If “zero Marketing” is outside or on the extreme edge of your training data, the model is **extrapolating** beyond typical observed ranges. This can reduce accuracy, especially if there are **nonlinear** relationships or thresholds (e.g., you need a minimum marketing spend to achieve certain profits). **Still a Valid Estimate** The model will provide a prediction, but the **confidence** might be lower if your training data has few (or no) examples with near-zero Marketing Spend.

10. **Coefficient Comparison** R&D: 0.845 Marketing: 0.016 R&D has a much larger marginal effect on profit per dollar spent. **Practical Considerations** This model suggests that **R&D** yields a higher return on investment. However, there may be **diminishing returns** in reality—beyond a certain point, extra R&D may not boost profit as much. Marketing may still be crucial for brand building, sales, etc. The model only captures **historical** relationships and may not reflect all strategic nuances.

11. **Potential Differences** Cost structures, labor markets, consumer behavior, and economic conditions can vary greatly by country. **Generalizability** A model trained on data from one region may **not** accurately reflect relationships elsewhere. You would ideally need **data from the target country** or a more diverse dataset that includes multiple countries to ensure the model is representative.

12. **Industry or Sector** (categorical): Different industries have different cost/profit structures. **Company Size** (number of employees, scale of operations). **Product/Service Type** (tech vs. manufacturing vs. services). **Location** (country, region, city)—operational costs vary widely. **Timing/Seasonality**: Some businesses see profit fluctuations based on seasons or economic cycles. **Pricing Strategy** or **Sales Channels**: Online vs. retail presence. **Quality or Brand Recognition**: Hard to quantify, but brand metrics might matter.

***STUDENT PERFORMANCE REGRESSION APPLIED***

1. In a linear regression model of the form:

* Each **coefficient** (β1,β2,…\beta\_1, \beta\_2, \dots) is a **slope** that indicates how much the **dependent variable** (Performance Index) changes when the corresponding **independent variable** increases by one unit, **holding all other variables constant**.
* The **intercept** (β0\beta\_0) is the predicted Performance Index when all independent variables are zero (often just a baseline value).

**Interpreting the Coefficients from Your Model**

1. **Hours Studied (2.852982):**
   * If a student studies **one additional hour**, their predicted Performance Index **increases by about 2.85 points**, on average.
2. **Previous Scores (1.018434):**
   * A **1-point increase** in previous test scores is associated with an **increase of about 1.02 points** in the Performance Index.
3. **Extracurricular Activities (0.612898):**
   * If a student is involved in extracurricular activities (Yes = 1 vs. No = 0), their Performance Index is **about 0.61 points higher** than a similar student not involved in extracurriculars.
4. **Sleep Hours (0.480560):**
   * An **additional hour** of sleep per day corresponds to a **0.48-point increase** in Performance Index, on average.
5. **Sample Question Papers Practiced (0.193802):**
   * Practicing **one additional sample question paper** is associated with a **0.19-point** increase in Performance Index.

2.The Performance Index is the outcome we want to predict, while the other five variables serve as inputs (features) to explain or forecast that outcome.

* **Independent (Predictor) Variables**:
  1. **Hours Studied**
  2. **Previous Scores**
  3. **Extracurricular Activities** (Yes/No)
  4. **Sleep Hours**
  5. **Sample Question Papers Practiced**
* **Dependent (Target) Variable**:
  1. **Performance Index**

3.Adding features can help if they are **meaningful** and **independent** but can harm the model if they introduce noise or multicollinearity.

1. **Potential Increase in Training Accuracy**
   * Adding more features often **increases R2R^2** (the model can fit the training data more closely).
2. **Risk of Overfitting**
   * If new features are **not truly predictive** (or if they are highly correlated with existing features), the model might memorize noise rather than learning generalizable patterns.
   * Overfitting shows up as **high training accuracy** but **poor performance** on unseen data (test set).
3. **Multicollinearity**
   * When features are highly correlated with each other, it can make coefficient estimates **unstable** and **hard to interpret**.
   * High Variance Inflation Factors (VIFs) are a sign of potential multicollinearity.

4.The **OLS Regression Results** provide **p-values** (under the “P>|t|” column) for each coefficient:

* + If **p < 0.05**, we typically consider the feature **statistically significant** at a 5% significance level.
  + In your summary, all features (Hours Studied, Previous Scores, Extracurricular Activities, Sleep Hours, Sample Question Papers Practiced) have very low p-values (essentially 0.000), indicating **all are statistically significant** predictors of Performance Index.

1. The **t-statistics** are also very high for Hours Studied and Previous Scores, reflecting their **strong influence** on Performance Index.
2. The **F-statistic** and corresponding p-value (0.00) indicate the overall model is statistically significant.

5. This discrepancy is a classic sign of **overfitting**. It means the model:

* Fits the **training data** extremely well (high R2R^2 on training set).
* Fails to **generalize** to new, unseen data (low R2R^2 or high error on test set).

Possible reasons include:

* **Too many features** relative to the number of observations, leading the model to learn noise.
* **Lack of representative training data** (the training set might not cover the variety in real-world scenarios).
* **Data leakage** or inadvertently including information in the training set that won’t be available in real-world predictions.

To address overfitting:

* Collect **more data** if possible.
* Use **regularization** techniques (e.g., Ridge, Lasso) to penalize overly complex models.
* Apply **cross-validation** to get more reliable estimates of out-of-sample performance.