

## Task 3: AI Logic & Explanation

### 1. Why this model was chosen.

Logistic Regression was chosen for this problem because:

- It performed better in Recall and F1-score compared to Random Forest, which is important for identifying churners.
- Churn prediction is a binary classification problem, well suited for Logistic Regression.
- The model is highly interpretable, allowing us to understand how each feature impacts churn.
- Feature coefficients provide clear direction (positive/negative) and strength of impact, which is useful for business decisions.

Logistic Regression provides a good balance between performance and interpretability, making it suitable for churn prediction.

### 2. How features impact prediction

Features that decrease churn:

- **Tenure:** Customers with longer tenure are much less likely to churn, showing strong customer loyalty over time.
- **Contract\_Two year & Contract\_One year:** Long-term contracts significantly reduce churn compared to month-to-month contracts.
- **OnlineSecurity\_Yes & TechSupport\_Yes:** Customers with added security and technical support services are more likely to stay.
- **PhoneService\_Yes:** Customers using phone services show lower churn, indicating deeper service engagement.

Features that increase churn:

- **InternetService\_Fiber optic:** Customers using fiber optic internet are more likely to churn, possibly due to higher cost or service expectations.
- **TotalCharges:** Higher total charges slightly increase churn probability, indicating price sensitivity.

- **StreamingTV\_Yes & StreamingMovies\_Yes:** Customers using streaming services show higher churn, often associated with high-usage, price-sensitive behaviour.

**Features related to customer commitment and support reduce churn, while high-cost or high-usage services increase churn risk.**

### **3. What improvements can be done.**

The current model is a strong baseline, but it can be improved by:

- Handling class imbalance using class weights or SMOTE.
- Hyperparameter tuning to improve model performance.
- Trying advanced models such as XGBoost or LightGBM.
- Feature engineering, such as grouping tenure into buckets or creating usage-based features.
- Using Pipelines and ColumnTransformer to automate preprocessing.
- Threshold tuning to optimize recall based on business priorities.