Telco customer churn - Step-by-Step Project Workflow

Step 1: Data Preprocessing

1. **Data Cleaning**: Handle missing values by either imputing or dropping them.
2. **Encoding Categorical Variables**: Use one-hot encoding for categorical features like contract type and payment method.
3. **Scaling**: Standardize or normalize numerical features to ensure compatibility with algorithms sensitive to feature scale (e.g., SVM and KNN).
4. **Train-Test Split**: Split data into training and testing sets (e.g., 80% train, 20% test).

Step 2: Exploratory Data Analysis (EDA)

1. Analyze feature distributions, correlations, and any potential relationships with the target variable.
2. Visualize relationships between numerical/categorical features and the churn target.

Step 3: Feature Engineering

1. **Interaction Features**: For example, create a feature for "Monthly Charges per Tenure" to assess usage consistency.
2. **Aggregation Features**: Generate features by grouping and summarizing data, like total usage and average monthly charge.

Step 4: Model Selection and Training

Train the following models and tune their hyperparameters:

1. **Logistic Regression**
   - Use it as a baseline model.
   - Evaluate the impact of regularization (L2 or L1).
2. **Decision Tree**
   - Tune depth and minimum sample split to optimize performance.
3. **Random Forest**
   - Tune parameters like the number of trees and maximum depth.
   - Use feature importance to interpret which factors contribute most to churn.
4. **XGBoost**
   - Perform hyperparameter tuning (learning rate, max depth, number of estimators).
   - Enable early stopping to prevent overfitting.
5. **Support Vector Machine (SVM)**
   - Choose between linear and RBF kernels and tune the C and gamma parameters.
   - Standardize the data before training to improve performance.
6. **Naive Bayes**
   - Use Gaussian Naive Bayes as a straightforward approach.
   - Note that it may work better on specific feature sets (e.g., after feature reduction).

7. **K-Nearest Neighbors (KNN)**
    - Tune the number of neighbors (k) and use a distance metric like Euclidean distance.
    - Scale features as KNN is sensitive to feature magnitude.

Step 5: Model Evaluation

1. **Metrics**: Evaluate models using accuracy, precision, recall, F1-score, and ROC-AUC score.
2. **Cross-Validation**: Use k-fold cross-validation to assess the robustness of each model and to prevent overfitting.

Step 6: Model Comparison

1. Compare the models based on their performance metrics.
2. Select the best-performing model based on recall or F1-score if minimizing false negatives is a priority (e.g., you don't want to miss identifying at-risk customers).

Step 7: Interpretability and Feature Importance

- For models like Random Forest, XGBoost, and Decision Tree, use feature importance to identify top predictors of churn.
- For Logistic Regression, examine coefficients to understand the direction of influence for each feature.

Step 8: Model Optimization and Ensemble Techniques (optional)

1. **Stacking or Blending**: Combine predictions from multiple models for potentially better performance.
2. **Ensemble Voting**: Use a hard or soft voting classifier with selected top models