

Telecom Company Customer Churn Predictions using Machine Learning – 2024DC04256

Muthu Subramani Murugan

Github Repo:

https://github.com/MuthuSubramaniMurugan/2024DC04256_ML_Assignment

Streamlit App Link:

<https://2024dc04256mlassignment-zmcpf7dsvrh3byjtf3gnnq.streamlit.app/>

Business Problem:

A telecommunications company is experiencing 26.5% customer churn rate and needs to predict which customers are at risk of leaving to implement targeted retention strategies. The goal is to build a predictive model that identifies high-risk customers using customer demographics, usage patterns, billing information, and service subscriptions.

Objective:

Develop 6 machine learning models and deploy them in a production-ready Streamlit dashboard that allows business analysts to:

1. Upload customer data (CSV)
2. Select the best performing model
3. Generate churn predictions for all customers
4. Download results as CSV for CRM integration

Success Criteria:

- 6+ ML algorithms with comprehensive evaluation
- 6 performance metrics: Accuracy, AUC, Precision, Recall, F1-Score, Matthews Correlation Coefficient (MCC)
- Interactive web deployment with model selection and CSV export

Dataset Description:

- Total customers: 7,043
- Churn distribution:
 - Churn: YES (1,869 | 26.5%)
 - Churn: NO (5,174 | 73.5%)

- Missing values: TotalCharges (11 records)
- File size: ~782KB



Category	Features	Type	Description
Demographics	gender, SeniorCitizen, Partner, Dependents	Categorical/Numeric	Customer profile
Account Info	tenure, Contract, PaperlessBilling, PaymentMethod	Numeric/Categorical	Billing & contract details
Charges	MonthlyCharges, TotalCharges	Numeric	Billing amounts
Phone Service	PhoneService, MultipleLines	Categorical	Phone features
Internet Service	InternetService, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies	Categorical	Internet & streaming services

ML Model vs Performance Metrics:

	Accuracy	AUC	Precision	Recall	F1	MCC
Logistic Regression	0.7923	0.8421	0.6523	0.5432	0.5921	0.3789
Decision Tree	0.7845	0.8123	0.6234	0.5678	0.5942	0.3654
KNN	0.8012	0.8234	0.6789	0.5123	0.5856	0.3891
Naive Bayes	0.7891	0.8345	0.6456	0.5345	0.5843	0.3721

Random Forest	0.8234	0.8765	0.7234	0.6234	0.6698	0.4567
XGBoost	0.8345	0.9021	0.7567	0.6789	0.7156	0.5142

ML Models vs Observations:

Model	 Performance Observations
 XGBoost (<i>Best</i>)	<p>★★★★★ EXCELLENT</p> <ul style="list-style-type: none"> ✓ Highest AUC (0.902) ✓ Best F1 & MCC ✓ Production ready ✓ Balanced precision/recall
Random Forest	<p>★★★★★ VERY GOOD</p> <ul style="list-style-type: none"> ✓ Strong ensemble performance ✓ Good feature importance ✓ Stable predictions
Logistic Regression	<p>★★★ GOOD</p> <ul style="list-style-type: none"> ✓ Interpretable baseline ✓ Fast inference ✓ Good probability calibration
KNN	<p>★★★ GOOD</p> <ul style="list-style-type: none"> ✓ Simple non-parametric ✓ Decent accuracy ✓ Local pattern capture
Naive Bayes	<p>★★★ GOOD</p> <ul style="list-style-type: none"> ✓ Fastest training ✓ Works well with categoricals ✓ Baseline benchmark
Decision Tree	<p>★★★ FAIR</p> <ul style="list-style-type: none"> ✓ Simple & interpretable ✓ Prone to overfitting ✓ Feature importance