# customer churn

# Project Objective

The objective of this project is to provide a comprehensive analytical solution for a startup e-commerce company. The solution includes building an enterprise data warehouse, performing data preprocessing and analysis, creating interactive dashboards, and implementing machine learning models for predicting customer churn. This end-to-end solution aims to help the company manage its data effectively, gain valuable insights, and make data-driven decisions to enhance customer satisfaction and reduce churn rates.

### Project Description

We undertook a project for a startup e-commerce company that lacked a comprehensive system to manage and track their increasing data. Previously, the company stored their data in various formats such as Excel, CSV, flat files, and other in-house tools. As their data grew, they sought to implement a centralized system to effectively manage and analyze their data. Our objective was to deliver an end-to-end analytical solution, including the creation of a centralized (enterprise) data warehouse, dashboards for performance tracking, and machine learning models for churn prediction.

### Data Analyst Role

As a Data Analyst, my role involved multiple key responsibilities throughout the project:

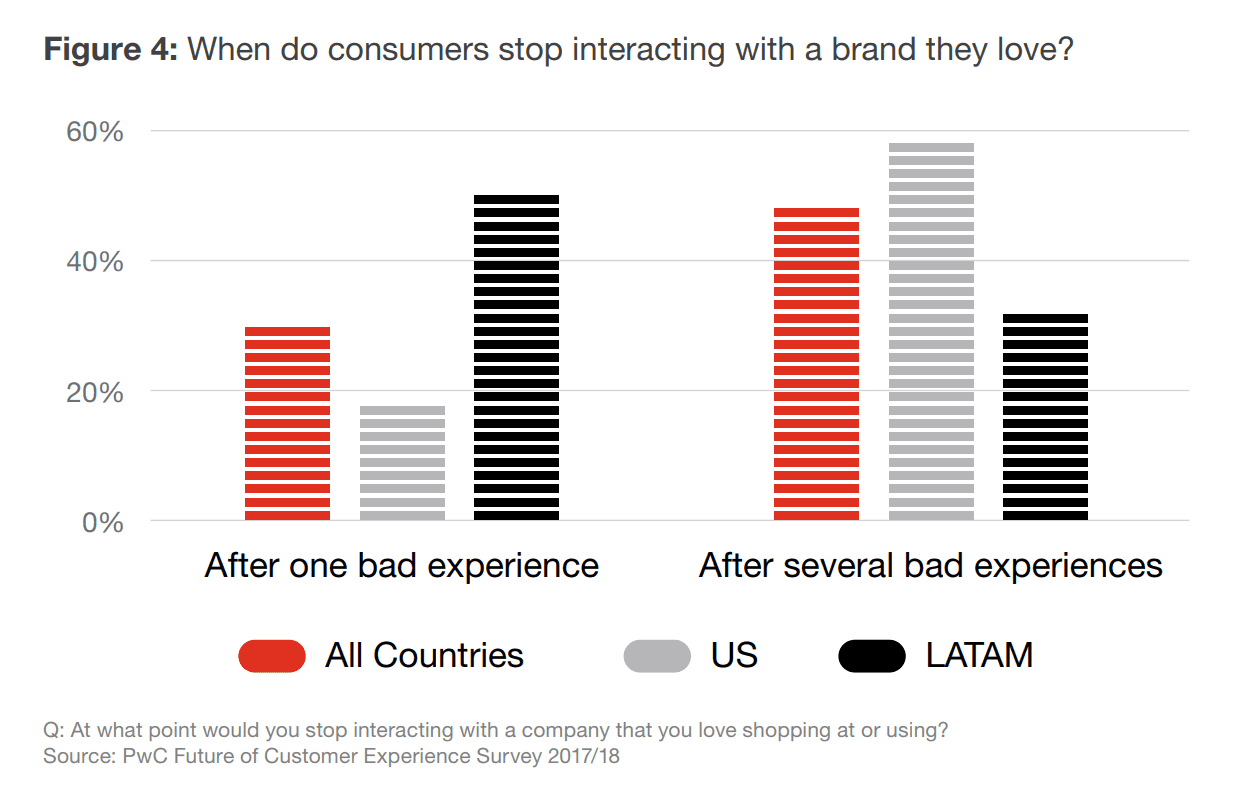
1. **Requirement Gathering and Documentation**:
   * Met with the client to understand their requirements and objectives.
   * Prepared technical documentation detailing the amount of data, data sources, and formats used by the client.
2. **Building the Enterprise Data Warehouse**:
   * Utilized SQL Server Integration Services (SSIS) to construct the ETL (Extract, Transform, Load) pipeline.
   * Extracted data from multiple sources, including Excel, CSV, and flat files.
   * Transformed the data to address anomalies and inconsistencies, as much of the data was manually entered.
   * Loaded the cleaned and transformed data into a SQL Server database.
3. **Data Preprocessing and Analysis**:
   * Conducted data preprocessing and analysis using Python.
   * Addressed data cleaning tasks such as handling null values and fixing anomalies.
   * Used libraries like Pandas, NumPy, Matplotlib, and Plotly for data analysis and visualization.
4. **Dashboard Creation**:
   * Built interactive dashboards using Tableau and PowerBI.
   * Created multiple dashboards to showcase various data aspects, including sales data, orders data, finance data, customer data, and product data.
5. **Customer Churn Prediction**:
   * Implemented machine learning models to predict customer churn.
   * Evaluated model performance and monitored predictions to ensure accuracy.

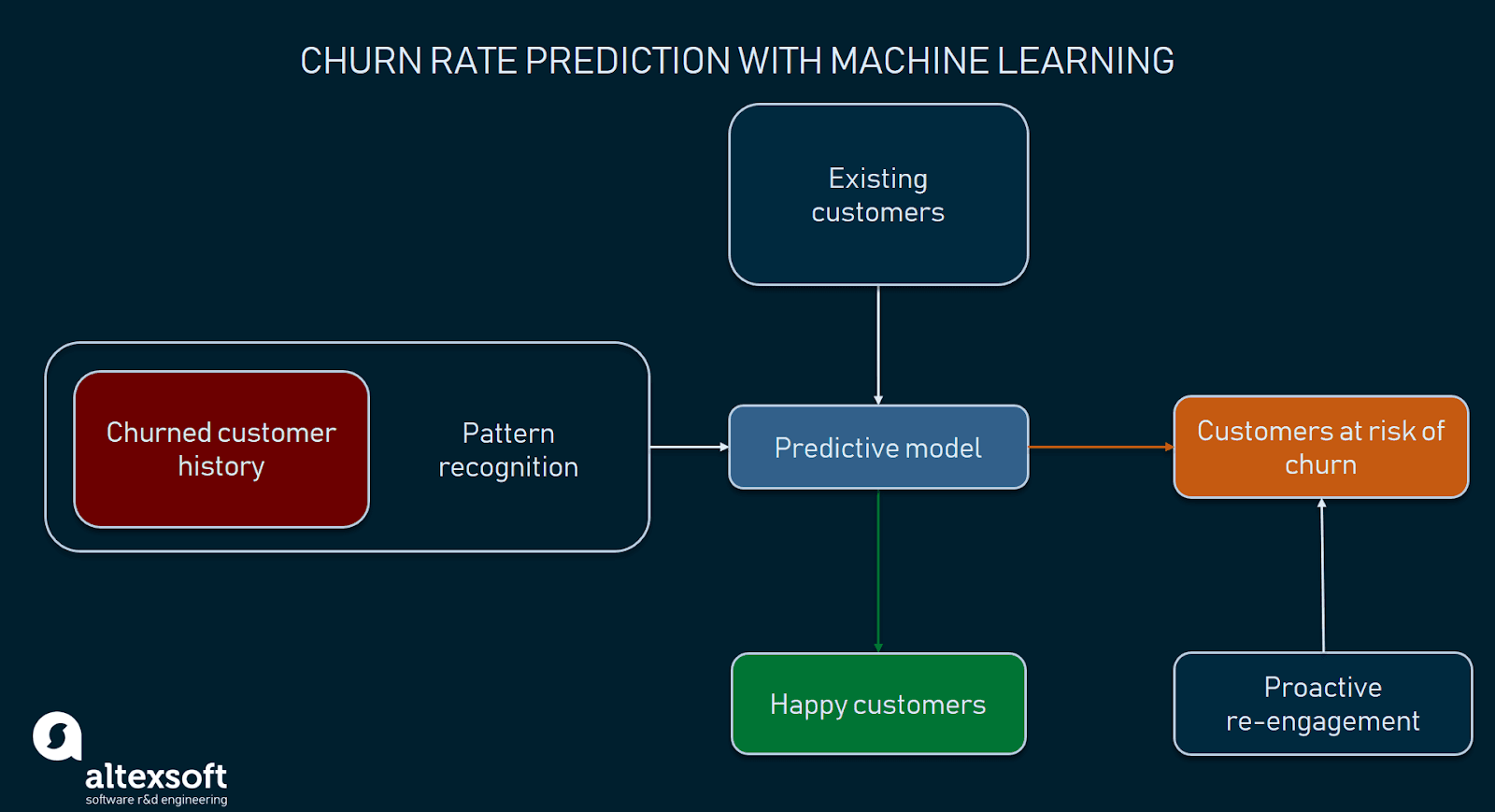
### Data Sources

* Sales data
* Orders data
* Finance data
* Customer data
* Products data
* Churn data

### Tools and Technologies Used

* **ETL Pipeline**: SSIS (SQL Server Integration Services)
* **Database**: SQL Server
* **Data Preprocessing and Analysis**: Python, Pandas, NumPy, Matplotlib, Plotly
* **Dashboard Creation**: Tableau, PowerBI
* **Machine Learning**: Scikit-learn





### **Predicting customer churn with machine learning**

But how to start working with churn rate prediction? Which data is needed? And what are the steps to implementation?

As with any machine learning task, data science specialists first need data to work with. Depending on the goal, researchers define what data they must collect. Next, selected data is prepared, preprocessed, and transformed in a form suitable for building machine learning models. Finding the right methods to training machines, fine-tuning the models, and selecting the best performers is another significant part of the work. Once a model that makes predictions with the highest accuracy is chosen, it can be put into production.

The overall scope of work data scientists carry out to build ML-powered systems capable to forecast customer attrition may look like the following:

* Understanding a problem and final goal
* Data collection
* Data preparation and preprocessing
* Modeling and testing
* Model deployment and monitoring

#### **Understanding a problem and a final goal**

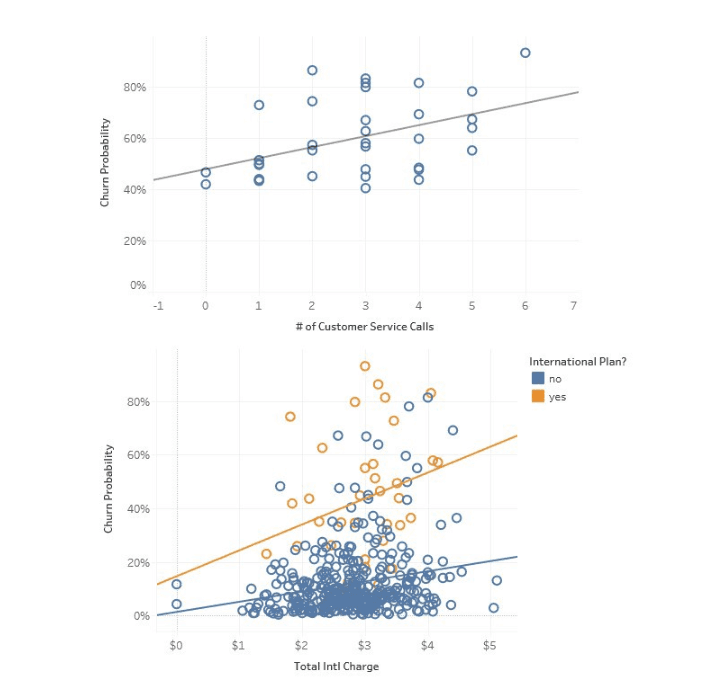
It’s important to understand what insights one needs to get from the analysis. In short, you must decide what question to ask and consequently what type of machine learning problem to solve: classification or regression. Sounds complicated, but bear with us.

**Classification.** The goal of classification is to determine to which class or category a data point (customer in our case) belongs to. For classification problems, data scientists would use historical data with predefined target variables AKA labels (churner/non-churner) – answers that need to be predicted – to train an algorithm. With classification, businesses can answer the following questions:

* Will this customer churn or not?
* Will a customer renew their subscription?
* Will a user downgrade a pricing plan?
* Are there any signs of unusual customer behavior?

The fourth question about atypical behavior signs represents a type of a classification problem called *anomaly detection*. Anomaly detection is about identifying outliers – data points that significantly deviate from the rest of the data.

**Regression.** Customer churn prediction can be also formulated as a regression task. Regression analysis is a statistical technique to estimate the relationship between a target variable and other data values that influence the target variable, expressed in continuous values. If that’s too hard – the result of regression is always some number, while classification always suggests a category. In addition, regression analysis allows for estimating how many different variables in data influence a target variable. With regression, businesses can forecast in what period of time a specific customer is likely to churn or receive some probability estimate of churn per customer.



**Identifying data sources.** Once you’ve identified which kinds of insights to look for, you can decide what data sources are necessary for further predictive modeling. Let’s assume the most common sources of data you can use for predicting churn:

* CRM systems (including sales and customer support records)
* Analytics services (e.g., Google Analytics, AWStats, CrazyEgg)
* Feedback on social media and review platforms
* Feedback provided on request for your organization, etc.

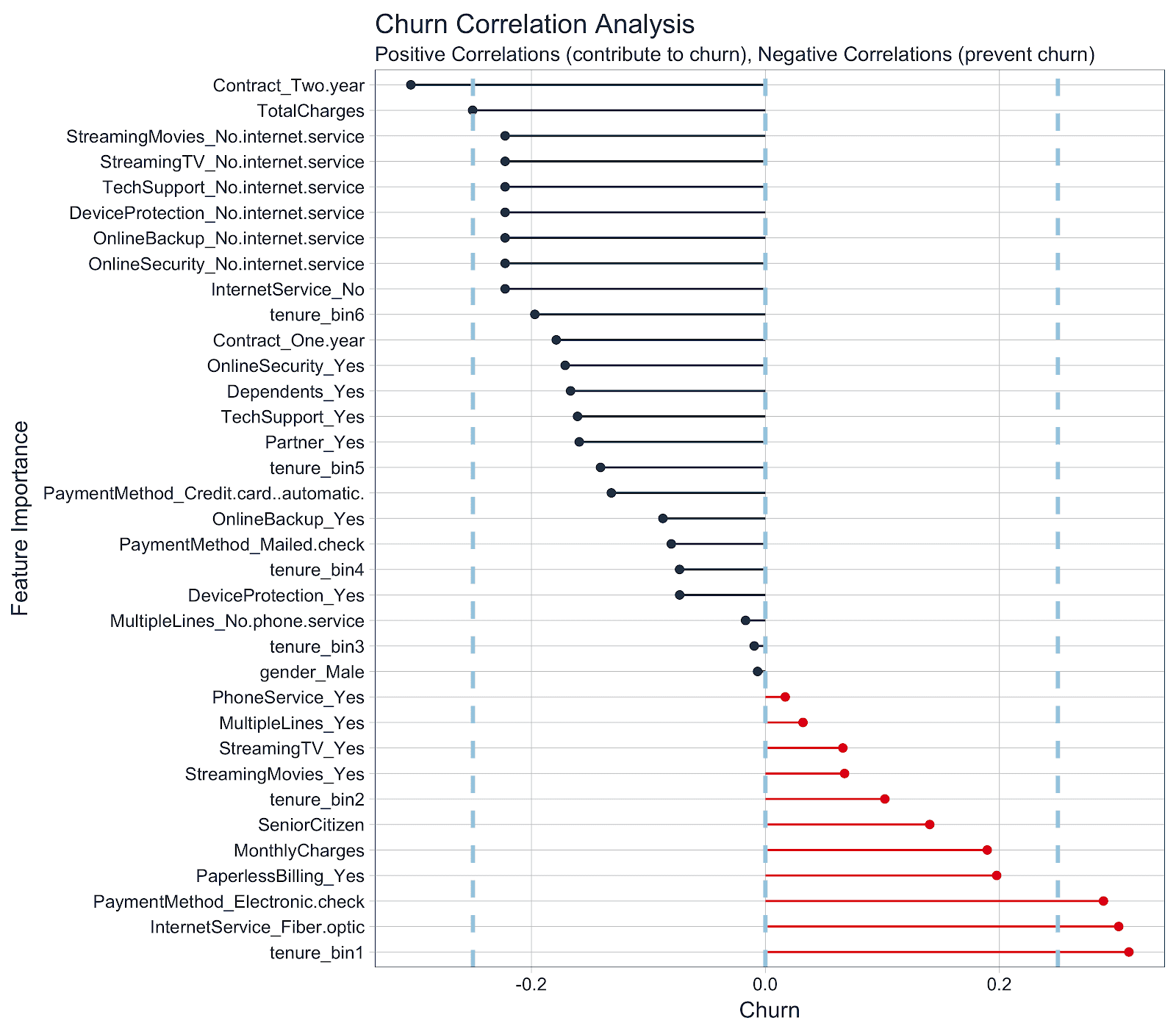
Obviously, the list may be longer or shorter depending on the industry.

#### **Data preparation and preprocessing**

Historical data that was selected for solving the problem must be transformed into a format suitable for machine learning. Since model performance and therefore the quality of received insights depend on the quality of data, the primary aim is to make sure all data points are presented using the same logic, and the overall dataset is free of inconsistencies. Previously we wrote an article about [basic techniques for dataset preparation](https://www.altexsoft.com/blog/datascience/preparing-your-dataset-for-machine-learning-8-basic-techniques-that-make-your-data-better/), so feel free to check it out if you want to know more on the topic.

**Feature engineering, extraction, and selection.** *Feature engineering* is a very important part of dataset preparation. During the process, data scientists create a set of attributes (input features) that represent various behavior patterns related to customer engagement level with a service or product. In a broad sense, features are measurable characteristics of observations that an ML model takes into account to predict outcomes (in our case the decision relates to churn probability.)

* **customer demographic features** that contain basic information about a customer (e.g., age, education level, location, income)
* **user behavior features** describing how a person uses a service or product (e.g., lifecycle stage, number of times they log in into their accounts, active session length, time of the day when a product is used actively, features or modules used, actions, monetary value)
* **support features** that characterize interactions with customer support (e.g., queries sent, number of interactions, history of customer satisfaction scores)
* **contextual features** representing other contextual information about a customer.



**Customer segmentation.**

Growing companies and those expanding their product range usually segment their customers using previously defined and selected features. Customers can be divided into subgroups based on their lifecycle stage, needs, used solutions, level of engagement, monetary value, or basic information. Since every customer category shares common behavior patterns, it’s possible to increase prediction accuracy through the use of ML models trained specifically on datasets representing each segment.

**Selecting an observation window (customer event history).** Predictive modeling is about learning the relationship between observations made during a period (window) that ends before a specific time point and predictions about a period that starts after the same time point. The former period is referred to

After data preparation, feature selection, and customer segmentation stages, the time comes to define how long it will take to track user behavior before drawing predictions.

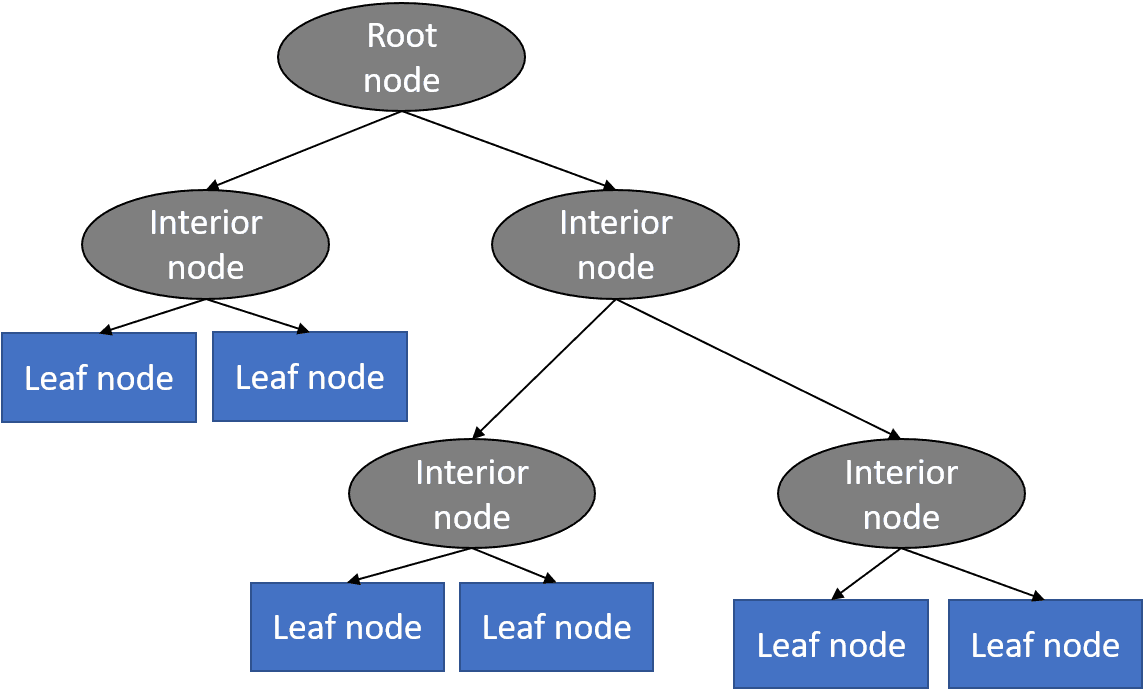
#### **Modeling and testing**

The main goal of this project stage is to develop a churn prediction model. Specialists usually train numerous models, tune, evaluate, and test them to define the one that detects potential churners with the desired level of accuracy on training data.

Classic machine learning models are commonly used for predicting customer attrition, for example, logistic regression, decision trees, random forest, and others. Alex Bekker from ScienceSoft suggests using Random Forest as a baseline model, then *“the performance of such models as XGBoost, LightGBM, or CatBoost can be assessed.”* Data scientists generally use a baseline model’s performance as a metric to compare the prediction accuracy of more complex algorithms.

[**Logistic regression**](https://machinelearningmastery.com/logistic-regression-for-machine-learning/) is an algorithm used for binary classification problems. It predicts the likelihood of an event by measuring the relationship between a dependent variable and one or more independent variables (features). More specifically, logistic regression will predict the possibility of an instance (data point) belonging to the default category.

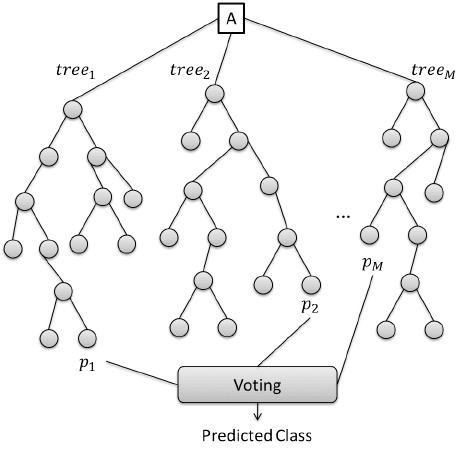
A [**decision tree**](https://www.analyticsvidhya.com/blog/2016/04/complete-tutorial-tree-based-modeling-scratch-in-python/) is a type of supervised learning algorithm (with a predefined target variable.) While mostly used in classification tasks, it can handle numeric data as well. This algorithm splits a data sample into two or more homogeneous sets based on the most significant differentiator in input variables to make a prediction. With each split, a part of a tree is being generated. As a result, a tree with decision nodes and leaf nodes (which are decisions or classifications) is developed. A tree starts from a root node – the best predictor.



**Decision tree basic structure. Source:** [**Python Machine Learning Tutorial**](https://www.python-course.eu/Decision_Trees.php)

Prediction results of decision trees can be easily interpreted and visualized. Even people without an analytical or data science background can understand how a certain output appeared. Compared to other algorithms, decision trees require less data preparation, which is also an advantage. However, they may be unstable if any small changes were made in data. In other words, variations in data may lead to radically different trees being generated. To address this issue, data scientists use decision trees in a group (AKA ensemble) that we’ll talk about next.

A **Random forest** is a type of an ensemble learning method that uses numerous decision trees to achieve higher prediction accuracy and model stability. This method deals with both regression and classification tasks. Every tree classifies a data instance (or votes for its class) based on attributes, and the forest chooses the classification that received the most votes. In the case of regression tasks, the average of different trees’ decisions is taken.



**XGBoost** is the implementation of the gradient boosted tree algorithms that’s commonly used for classification and regression problems. Gradient boosting is an algorithm consisting of a group of weaker models (trees), which sums up their estimates to predict a target variable with more accuracy.

A group of researchers from the University of Virginia studied the time-dependent software feature usage data, such as login numbers and comment numbers, to predict a SaaS customer churn within the time horizon of three months. The authors compared model performance across four classification algorithms, and *“the XGBoost model achieved the best results for identifying the most important software usage features and for classifying customers as either churn type or non-risky type.”* The XGBoost model’s ability to define the most significant features that represent how customers use SaaS software can help service providers launch more effective marketing campaigns when targeting potential clients, according to researchers.

[**LightGBM**](https://lightgbm.readthedocs.io/en/latest/) is a gradient boosting framework that uses tree-based learning algorithms. It can be used for many ML tasks, for instance, classification and ranking. According to the documentation, some advantages of LightGBM are faster training speed and higher efficiency, as well as greater accuracy. These algorithms use lower memory and handle large volumes of data – it’s not advisable to use them on datasets with less than 10,000 rows. LightGBM also supports parallel and GPU learning (the use of graphical processing units for training large datasets).

[**CatBoost**](https://github.com/catboost) is another gradient boosting on decision trees library. It handles both numerical and categorical features, so can be used for classification, regression, ranking, and other machine learning tasks. One of the pros of CatBoost is that it permits training models with CPU and two or more GPUs.

#### **Deployment and monitoring**

And now, the final stage of the churn prediction project workflow. The selected model/models need to be put into production. A model may be incorporated into existing software or become a core of a new program. However, the deploy-and-forget scenario won’t work: Data scientists must keep track of a model’s accuracy levels and improve it if needed.

### **Conclusion**

Churn rate is a health indicator for subscription-based companies. The ability to identify customers that aren’t happy with provided solutions allows businesses to learn about product or pricing plan weak points, operation issues, as well as customer preferences and expectations to proactively reduce reasons for churn.

It’s important to define data sources and observation period to have a full picture of the history of customer interaction. Selection of the most significant features for a model would influence its predictive performance: The more qualitative the dataset, the more precise forecasts are.

Companies with a large customer base and numerous offerings would benefit from customer segmentation. The number and choice of ML models may also depend on segmentation results. Data scientists also need to monitor deployed models, and revise and adapt features to maintain the desired level of prediction accuracy.