Trinamix

Forecasting Using Machine Learning Algorithms in Fusion Demand Management

Blog By – Sandeep Pati

Oracle’s patented Bayesian analytical forecast engine applies cross-validation machine learning and an ensemble of 15 industry-standard and proprietary statistical models to predict demand for both continuous and intermittent data series.

Machine learning (ML), a subset of artificial intelligence involves showing a large volume of data to a machine so that it can learn and make predictions, find patterns, or classify data.

Everyone learns differently–including machines! Generally, data scientists use different learning styles to train machine learning algorithms. It is mainly classified into 4 types:

1. **Supervised Learning**

The objective is to train the algorithm to predict accurate labels for new, unseen data. This type of machine learning is called supervised learning because the algorithm is provided with labeled data to learn from.

Supervised learning is a type of machine learning algorithm where the system is trained using a dataset that consists of both the input data and the corresponding correct output or target. During the training process, the algorithm learns to identify patterns, relationships, and features within the data so that it can map the input data to the output effectively.

2. **Unsupervised Learning**

The objective is to analyze unlabeled data (Raw Observations) without predefined output labels and to discover patterns, relationships, or structures within the data.

3. **Semi-supervised Learning**

It is a hybrid approach that combines labeled and unlabeled data for machine learning. The method uses limited labeled data and a larger set of unlabeled data to improve the learning process.

4. **Reinforcement Learning**

It is a machine-learning algorithm inspired by how humans learn from trial and error. It enables machines to learn from their experiences.

Oracle Demand Management machine learning forecasting methods categorized into 4 Different Types

Regression-based models: Supervised ML Algorithms-

Supervised machine learning involves training a model on labeled data, where the input features (In machine learning, features refer to the inputs or attributes used to train a model) and their corresponding targets feature (simply the "target," is the specific variable or attribute that a machine learning model is designed to predict or estimate based on input data) are known. The model learns from this labeled data to make predictions or classify new, unseen data points. Here are some of the Models based on this:

1. Regression
2. Regression For Intermittent
3. Transformation Regression
4. Modified Ridge Regression
5. Multiplicative Monte Carlo Regression
6. Multiplicative Monte Carlo Intermittent Data
7. Logistic Regression
8. Auto-Regressive Logistic
9. Combined Transformation
10. Dual Group Multiplicative

Classical Heuristic Model:

Unlike Supervised, Heuristics are general strategies or rules of thumb used to solve problems or make decisions, often based on experience or intuition rather than explicit mathematical models. In Layman's Terms -It is like an intelligent guess or shortcut to finding solutions.

Classical heuristic models are helpful when you need to make decisions quickly, and you do not have the time or resources (Supporting Predictors) to find the absolute best solution. Some methods Include:

1. Croston For Intermittent
2. Holt Exponential Smoothing

Enhanced classical models (ECM):

The enhanced classical model (ECM) in machine learning is a variation of the classical linear regression model that allows for non-linear relationships between the features and target variables. This model uses a non-linear function, such as a polynomial or radial basis function, to transform the data before fitting a linear regression model.

1. Causal Winters
2. Auto-Regressive Integrated External (ARIX)
3. Auto-Regressive External Inputs (ARX)

Feature-based model: eXtreme Gradient Boosting (XBG)

Extreme Gradient Boosting is a tree-based algorithm, which sits under the supervised branch of Machine Learning.

It is a popular machine-learning technique used for classification and regression tasks. It is an iterative algorithm that combines multiple weak models to create a strong predictor.

XGB stands out for its ability to handle large datasets, high-dimensional feature spaces, and complex models with non-linear interactions.

Model 1: Linear Regression:

In simple terms, linear regression takes a set of data points with known input and output values and finds the line that best fits those points. This line, known as the "regression line," serves as a predictive model. By using this line, we can estimate or predict the output value (Y) for a given input value (X). It is represented by a linear equation **Y= m *X + c**.

Let's take an example of Pharmaceuticals Business to forecast how many people will buy the medicine next month based on how much they spend on advertising. They think there's a relationship between advertising and sales.

- Y (Dependent Variable) represents the number of people buying the medicine (what we want to predict).
- X (Independent variable) represents the advertising expenses (our input or predictor).
- "m" is the slope, which tells us how much the number of people buying the medicine changes for each dollar spent on advertising.
  m = Σ[(X - Mean(X)) * (Y - Mean(Y))] / Σ[(X - Mean(X))^2]
- "c" is the intercept, which is like a starting point for our prediction.
  c = Mean(Y) - a * Mean(X)

Model 2: Regression For Intermittent:

Imagine a company that makes and sells VR glasses. They want to predict how many VR glasses they will sell in the future, but here's the challenge: they don't have sales data for every single day because people don't buy VR glasses every day. Some days, they sell a lot, and on others, they sell none.

Sales Data: Oracle has data (Basically Bookings or Shipments History) on how many VR glasses were sold on certain days. For example:

- Monday: 5 glasses sold
- Tuesday: 8 glasses sold
- Wednesday: No glasses sold
- Thursday: No glasses sold
- Friday: 10 glasses sold

Predicting Future Sales: Now, Oracle wants to predict how many VR glasses they might sell next month, even on those days when they don't have data (like Wednesdays and Thursdays).

The intermittent model is like a special tool that looks at the days they do have data for and tries to find patterns:

- It is noticed that Fridays tend to have more sales, maybe because people have more free time on weekends.

- It also checks if there were any special promotions or events that boosted sales on certain days.

Model 3: Transformation Regression:

A Transformation Regression Model in Oracle Demand Management Cloud helps companies figure out how different factors influence the sales of the Business.

Let's use an Example: *Predicting Snack Sales for a Chip Company*

Imagine a chip company running DM Cloud, and he/she wants to know how different factors influence your chip sales. You collect data on various things:

- Price: The cost of a bag of chips.
- Time of Year: Whether it's summer, winter, or another season.
- Advertising Spending: How much money you spend on ads for your chips.

Now, you use the Transformation Regression Model:

Step 1: You might turn the season into numbers (like 1 for summer, 2 for winter) and scale down the price and advertising spending to make them easier to work with.

Step 2: The Regression Analysis model looks at your data and finds patterns. For instance, it might be noticed that chip sales tend to go up in the summer when it's hot and people are outside more.

Step 3: The simplified mathematical model could look like this:

Chip Sales = (Price Coefficient * Price) + (Season Coefficient * Season) + (Advertising Coefficient * Advertising Spending) + Intercept

- "Price Coefficient" shows how changing the price affects sales.
- "Season Coefficient" shows how different seasons impact sales.
- "Advertising Coefficient" shows how advertising spending influences sales.
- "Intercept" accounts for other factors that affect sales but aren't in the model.

Step 4: Prediction With this model, you can predict future chip sales. For instance, if you plan to lower the price in the summer and increase your advertising spending, you can use the model to estimate how many bags of chips you might sell.

So, in simple terms, the Transformation Regression Model helps companies like the chip company understand how different things (price, season, advertising) affect their sales by using math. It turns data into useful predictions, helping businesses make smart decisions.

Model 4: Modified Ridge Regression:

Modified Ridge Regression is used when there are lots of input factors (like price, advertising, and more) that could influence the output (like product sales).

What makes Modified Ridge Regression special is that it's good at preventing overfitting. Overfitting is like making a recipe too complicated. It happens when a regression model becomes too focused on the data it already has and can't predict new data well. It does this by adding a "penalty" to the mathematical equation that keeps the model from getting too complicated. This penalty prevents the model from giving too much importance to any one ingredient (or factor) and helps it make better predictions.

Let's take an Example: *Predicting Demand for Umbrellas*

Imagine you work for a company that sells umbrellas. Your goal is to predict how many umbrellas you'll sell on different days. But it's not just about rain; it's also about other things like the price of umbrellas, advertising, and the day of the week. Here's how the Modified Ridge Regression Model helps:

Step 1: You collect data for several months, including:

- The number of umbrellas sold each day.
- The price of umbrellas.
- How much money you spend on advertising.
- The day of the week (e.g., Monday, Tuesday, etc.).

Step 2: The Modified Ridge Regression Model looks at all this data and tries to figure out how each factor (price, advertising, day of the week) affects umbrella sales. How Much importance is to be given to each factor.

Step 3: Mathematical Model: The model creates a mathematical equation like

Umbrella Sales = (Price Coefficient * Price) + (Advertising Coefficient * Advertising Spending) + (Day Coefficients * Day of the Week) + Intercept

- "Price Coefficient" shows how changes in the umbrella's price affect sales.
- "Advertising Coefficient" shows the impact of advertising spending on sales.
- "Day Coefficients" account for how different days of the week influence sales.
- "Intercept" is like a starting point for sales.

Step 4: This Equation can be used to predict future umbrella sales. For example, if you plan to reduce the price and increase advertising spending on a rainy Friday, the model can help you estimate how many umbrellas you might sell.

Both *Transformation Regression & Modified Ridge Regression* may look similar in the Mathematical Model but the Use cases are different from one another.

Transformation Regression manipulates the original data by applying mathematical transformations like logarithms, square roots, or other functions to make it more linear or conform to the assumptions of linear regression whereas *Modified Ridge Regression* used when you have many predictor variables, and you want to ensure that the model remains stable and prevents overfitting, especially in situations where multicollinearity (high correlation between predictor variables) is present.

Model 5: Multiplicative Monte Carlo Regression:

The Multiplicative Monte Carlo Regression Model used in Oracle Demand Management Cloud is a sophisticated forecasting method that combines multiple factors in a multiplicative manner to generate predictions. This refers to the way the model combines different factors. Instead of simply adding them together (which would be an additive model), it multiplies them. This can capture more complex relationships between variables. This is a technique used to understand the impact of risk and uncertainty in prediction and forecasting models. It uses random sampling and statistical analysis to compute the results of uncertain parameters.

Let’s take an Example:

You’re trying to predict the sales of ice cream in a city. The Multiplicative Monte Carlo Regression Model might consider factors like temperature and whether it’s a weekend or a weekday.

Instead of just adding these factors together, it multiplies them, because maybe ice cream sales increase exponentially with temperature, and maybe weekends amplify this effect even more.

Mathematically, if we have two groups of causal factors (for example, temperature and day of the week), the Mathematical model is like:

Sales = (sum of temperature factors) * (sum of day-of-week factors)

Model 6: Multiplicative Monte Carlo Intermittent Data:

The Multiplicative Monte Carlo Regression for Intermittent (MMCR-I) is a type of advanced forecasting model used in Oracle Demand Management Cloud.

It’s typically used in scenarios where you have a large number of causal factors (like price, seasonality, promotional activities, and so on) and intermittent data - that is, there are periods where the demand is zero.

Model 7: Logistic Regression:

Also known as "logit regression," is a supervised learning algorithm primarily used for binary classification tasks.

Logistic regression predicts the probability that an input can be categorized into a single primary class. However, in practice, it is commonly used to group outputs into two categories: the primary class and not the primary class. To accomplish this, logistic regression creates a threshold or boundary for binary classification (Like Yes or No).

Let’s take an Example: *Predicting Product Demand*

Imagine a company that sells laptops. You want to know if a customer who visits your website will buy a laptop or not. You collect data on several factors like the customer's age, the laptop's price, and whether they clicked on an ad.

Here's how Logistic Regression works:

Step 1: You gather data about customers who visited your website, whether they bought a laptop (1 for yes, 0 for no), and factors like age, laptop price, and ad click (1 for yes, 0 for no).

Step 2: The model analyzes the data and finds patterns. It figures out how these factors affect the chances of someone buying a laptop.

Step 3: It uses a mathematical model called the logistic function to calculate the probability of an event happening (in this case, buying a laptop). The model looks something like this:

Probability of Buying Laptop = 1 / (1 + e^-(a * Age + b * Price + c * AdClick + d))

- "a," "b," "c," and "d" are numbers that the model calculates to fit the data. The model finds the best values for "a," "b," "c," and "d" to make accurate predictions based on the data.
- "e" is a mathematical constant.

Step 4: With this model, you can predict the probability of a customer buying a laptop based on their age, the laptop's price, and whether they clicked on an ad.

For example, it might tell you there's a 70% chance a 25-year-old customer will buy a laptop if the price is 40K and they clicked on an ad.

Model 8: Auto-Regressive Logistic:

An Auto-Regressive Logistic Model is a forecasting tool that's used when you want to predict binary outcomes (like yes/no, 1/0) *over time*, such as whether a customer will make a purchase or not. In simple words, it combines time-based patterns with logistic regression to make predictions.

Example: *Predicting Customer retention for an OTT Company*

A subscription-based streaming service wants to predict whether a customer will cancel their subscription in the next month (churn) based on their past behavior.

Step 1: You collect data for each customer over several months, including whether they churned (1 for churned, 0 for not) and their past behavior, like how many times they watched shows in the previous months.

Step 2: The "auto-regressive" part means you look at a customer's past behavior and use it to predict their future behavior. For instance, if a customer watched fewer shows in the past few months, that might indicate they're losing interest. It's like fitting a curve to the data that helps you predict the probability of a customer churning based on their past actions.

Step 3: People's behavior can change over time, so you account time factor in the model. For example, if a customer watched fewer shows a few months ago but then increased their viewing recently, it might indicate they're still interested.

Step 4: Using this model, you can predict the probability of a customer churning in the next month. If the probability is high, you might take actions like offering them discounts or suggesting new content to keep them engaged.

Model 9: Combined Transformation:

A Combined Transformation Model for Forecasting is a way of using different pieces of information to make predictions.

Example: *Predicting Flat Prices*

A real Estate company wants to predict how much a flat will sell for in a neighborhood before building a new project in a certain locality. To do this, you gather various types of information:

- Size of the House: The square footage of the flat.
- Number of Bedrooms: How many bedrooms the flat has.
- Location/proximity: The neighborhood where the Society is.
- Local School Ratings: How good the schools are in the area.
- Recent Home Sales: The prices at which similar flats in the same area recently sold.

Some of this data need to be transformed to apply a Combined Transformation Model. For example, you might convert the neighborhood names into numerical values, like 1 for "Downtown" and 2 for "Suburbs." You might also scale the size of the house and the number of bedrooms to be on a similar scale.

Now, you have all this transformed data combined in one place, like a big table. Each row in the table represents a house; you have information about its size, bedrooms, location, school ratings, and recent sales prices.

This supervised machine learning algorithm is like a regression model to learn from this combined data. The model looks at the historical data (the house features and the sale prices) to understand how these factors influence house prices.

Once the model has learned from the historical data, prices of new houses in the same neighborhood can be predicted. For example, if you input the size, number of bedrooms, location, and school ratings of a new house, the model can estimate its price based on what it learned from the combined data.

Model 10: Dual Group Multiplicative

This model is particularly useful when there are two distinct groups of factors that influence the forecast, and these factors interact with each other in a multiplicative (as opposed to additive) manner.

This refers to the two distinct groups of factors that the model considers. For example, these could be internal factors (like pricing or marketing efforts) and external factors (like economic indicators or weather patterns).

For example, imagine you're trying to forecast ice cream sales. One group of factors could be internal factors like price and advertising spend, and the other group could be external factors like temperature and whether it's a holiday. The Dual Group Multiplicative model would consider how all these factors interact. For instance, perhaps advertising has a bigger impact on sales when it's hot outside, or maybe price changes have a bigger effect during holidays.

Mathematically, if we have two groups of causal factors (for example, internal and external), the model might look something like this

Sales = (sum of internal factors) * (sum of external factors)

Model 11: Holt Exponential Smoothing:

The Holt Exponential Smoothing model considers the current sales level, the recent trend, and the smoothing parameters (α and β) to make these forecasts.

smoothing parameters that control the weight given to the current value and the previous level and trend, respectively. These values are typically between 0 and 1, and you can adjust them based on the data and your preferences.

Now, following formulas can be used to forecast future values:

- Level at Time (Lt): $Lt = \alpha * \text{Current Value} + (1 - \alpha) * (Lt\text{-}1 + Tt\text{-}1)$
- Trend at Time (Tt): $Tt = \beta * (Lt - Lt\text{-}1) + (1 - \beta) * Tt\text{-}1$
- Forecast for Time (Ft): $Ft = Lt + Tt$

Model 12: Croston For Intermittent:

Croston Model helps you predict sales for products that don't sell regularly. It does this by looking at past sales, figuring out how often they happen, and estimating how much you might sell in the future during those occasional sales periods.

Model 13: Causal Winters:

A time series forecasting technique used to predict future values of a time series that exhibits both a seasonal pattern and a trend. This method is useful when you want to forecast data points over time, considering both long-term trends and seasonal fluctuations.

Winter's method consists of three components: level (Lt), trend (Tt), and seasonality (St). Here's how to apply the Winter's method to forecast ice cream sales:

Use the following formulas to forecast future values:

- Level at Time (Lt): $Lt = \alpha * (\text{Sales at Time } (t)) + (1 - \alpha) * (Lt\text{-}1 + Tt\text{-}1)$
- Trend at Time (Tt): $Tt = \beta * (Lt - Lt\text{-}1) + (1 - \beta) * Tt\text{-}1$
- Seasonality at Time (St): $St = \gamma * (\text{Sales at Time } (t)) / Lt + (1 - \gamma) * St\text{-}m$, where m is the number of seasons .
- Forecast for Time (Ft): $Ft = (Lt + Tt) * St\text{-}m\text{+}t$, where m is the number of seasons.

Here, α, β, and γ are smoothing parameters between 0 and 1, which you can adjust based on the data and your preferences.

Model 14: Auto-Regressive External Inputs (ARX):

The ARX model is a type of linear regression model that uses a combination of autoregressive terms (AR) and exogenous inputs (X). The AR part indicates that the evolving variable of interest is regressed on its own lagged (i.e., prior) values. The X part refers to the exogenous inputs, which are external variables that influence the variable of interest.

Model 15: Auto-Regressive Integrated External (ARIX):

The ARIX model is a variant of the ARX model. It includes integrated auto-regression terms at lag 1 and an unknown seasonal lag k, and linear causal factors. The value of k is chosen from a set of possible seasonal indexes to produce the best fit. Causal factors include the constant and events (without seasonal causal factors and without time)

Model 16: Gradient Boosting (XBG):

Forecasting Profile to be used: Feature-Based Bookings/Shipments Forecast for New Products.

The eXtreme Gradient Boosting (XGB) model is a supervised learning method. Demand Management's built-in Planning Advisor uses machine learning to understand the relative importance of features in a new product versus those of existing products and generates forecasts based on the new product's features.

Gradient boosting algorithms employ an ensemble method, which means they create a series of "weak" models that are iteratively improved upon to form a strong predictive model. The iterative process gradually reduces the errors made by the previous models, leading to the generation of an optimal and powerful Predictive Model.

This Model evaluates up to 500 static and dynamic attributes to find the most predictive features without overfitting.

Please Note the above example gives a basic idea of how it works. The actual model used by Oracle Demand Management Cloud can be more complex, involving many variables and advanced math, but

About Author: -

Sandeep Pati is Dedicated and seasoned Supply Chain Planning Consultant with over 6 years of extensive experience in crafting transformative solutions across diverse industries. Specializing in Oracle Supply Chain Planning modules, I have successfully navigated the intricacies of Supply Planning, Demand Planning, Replenishment Planning, and Sales and Operation Planning.

References:

1. https://www.oracle.com/a/ocom/docs/applications/scm/oracle-supply-chain-planning-solution-brief.pdf
2. https://www.oracle.com/oce/dc/assets/CONTA3394DD16B544953A43D63EA2EBE040E/native/scm-scp-demandmanagement-datasheet-20c.pdf
3. Technical Brief / Oracle Fusion Cloud Demand Management – Analytical Forecasting Methods / Version 2.1
4. Oracle® Demantra Implementation Guide- Theoretical Engine Models https://docs.oracle.com/cd/E18727_01/doc.121/e05136/T485796T489188.htm-
5. Using Demand Management - Forecasting Methods https://docs.oracle.com/en/cloud/saas/supply-chain-and-manufacturing/23b/fasdm/forecasting-methods.html