Data Science Toolset

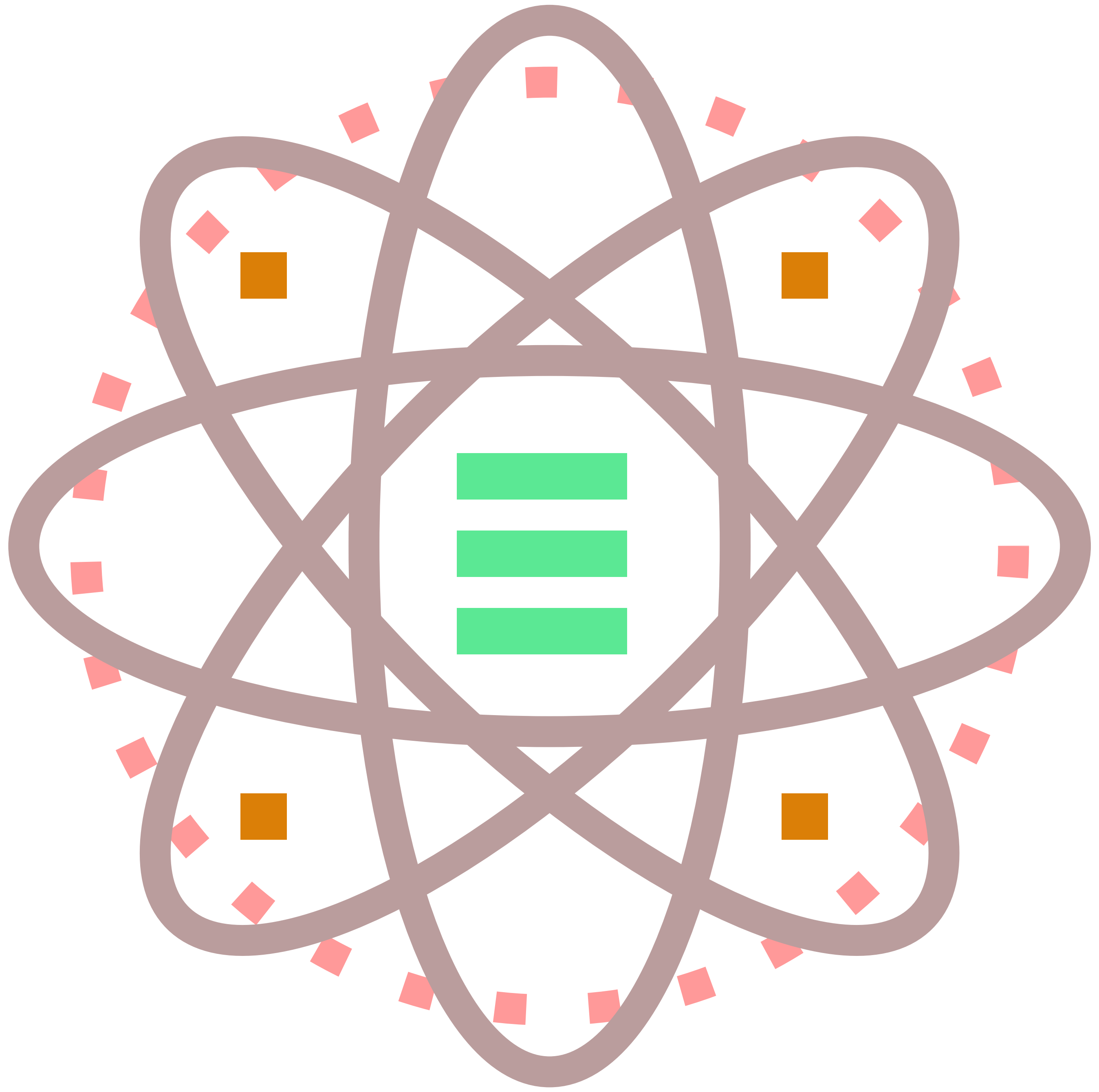
(Formerly known as Advanced Ads Toolset)

(Formerly known as Data Science Suite)

xandr

Technical Communications

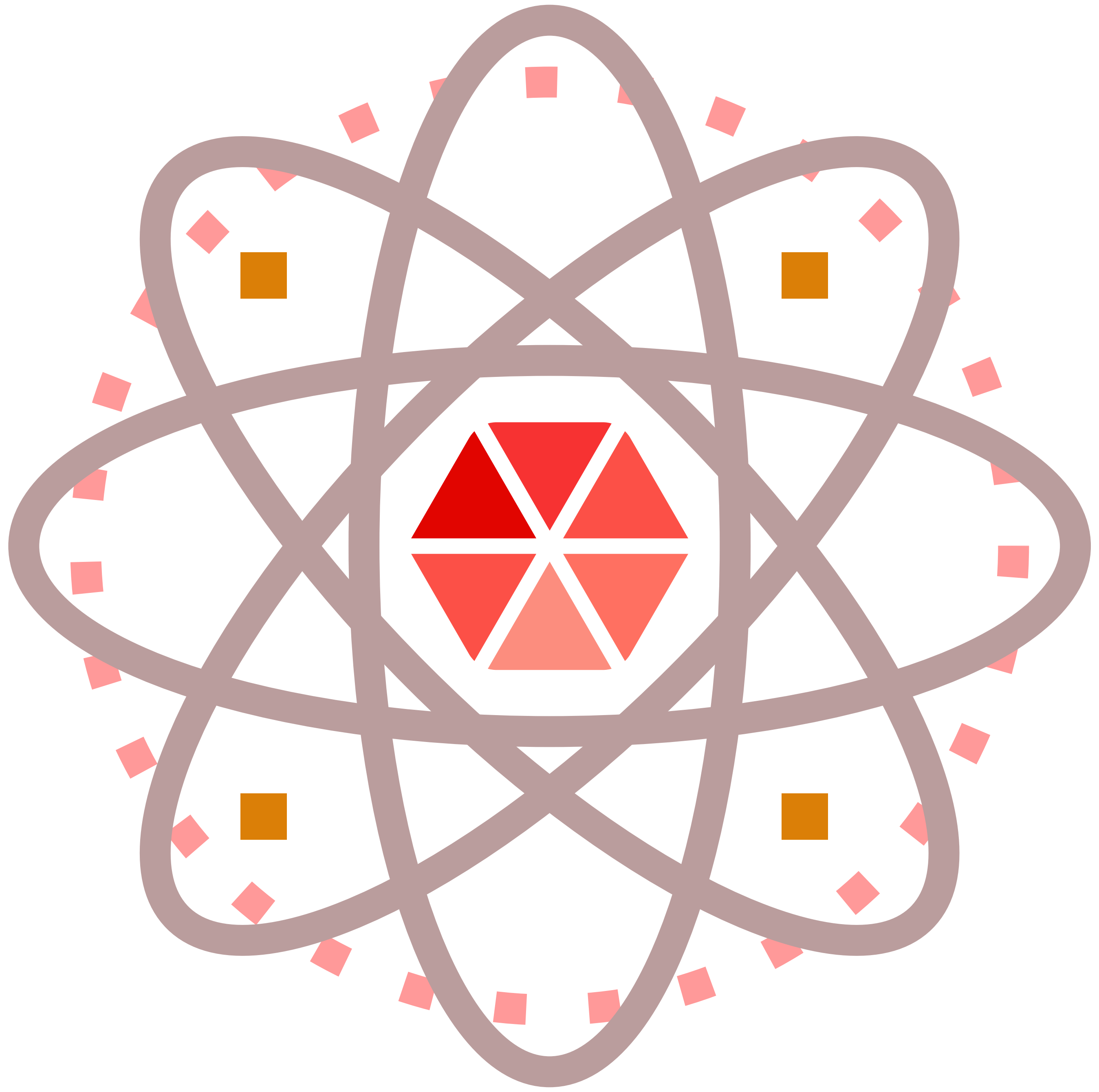
What is Data Science



A process of using scientific methods, algorithms, statistics, probability, and data systems to extract knowledge and insights from structured and unstructured data.

Simplified, data science analyzes large sets of data to provide meaningful information that can be used in decision making.

What is Data Science at Xandr



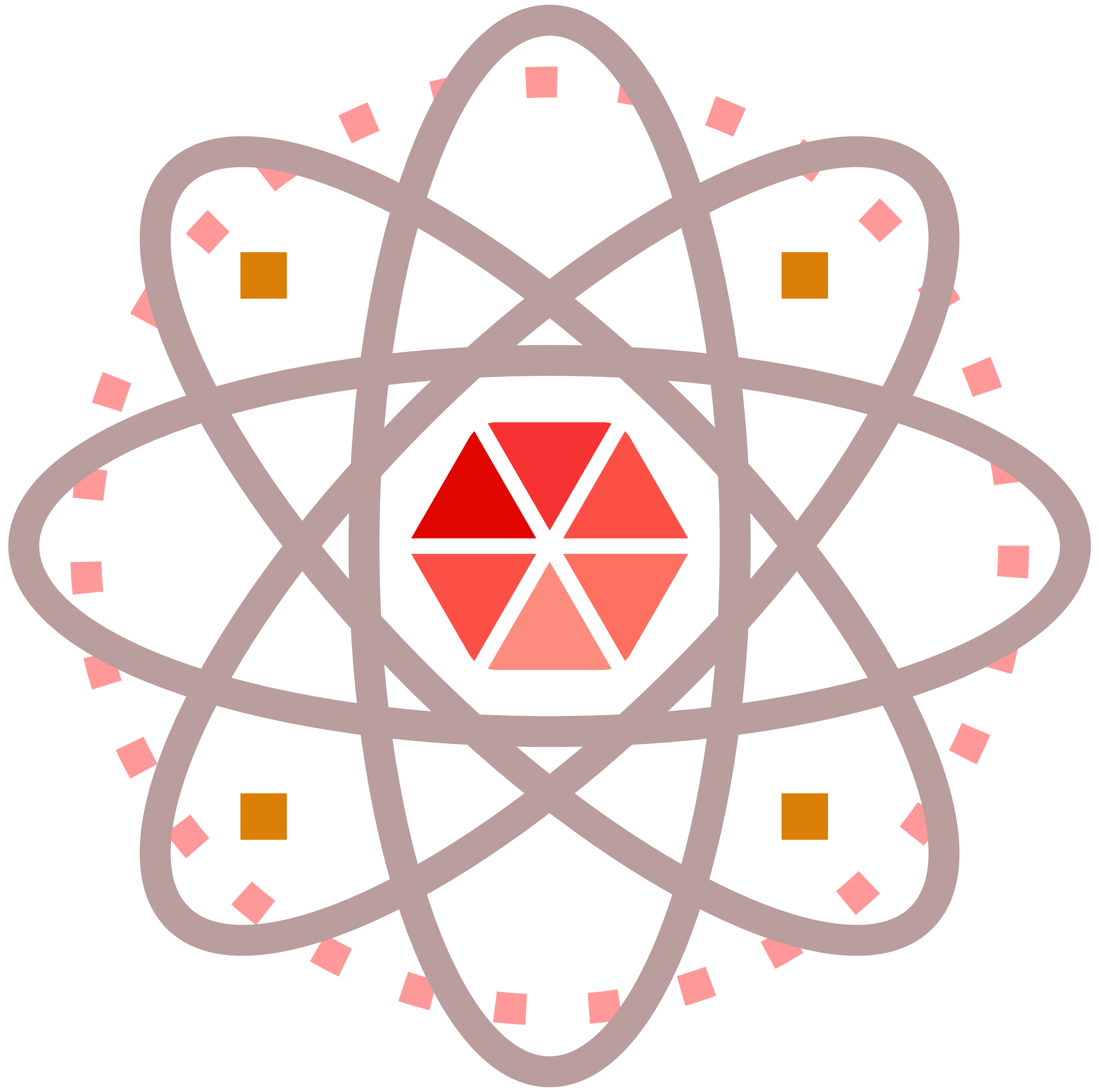
Data science at Xandr is used primarily to:

- Enhance bidding strategies with precise targeting.
- Improve measurement and testing capabilities.
- Create in-depth outcome analysis.

To achieve this our clients, with and without data scientist on their staff, can create Custom Models, utilize the Incrementality Feed, and access the User Group Pattern Service.

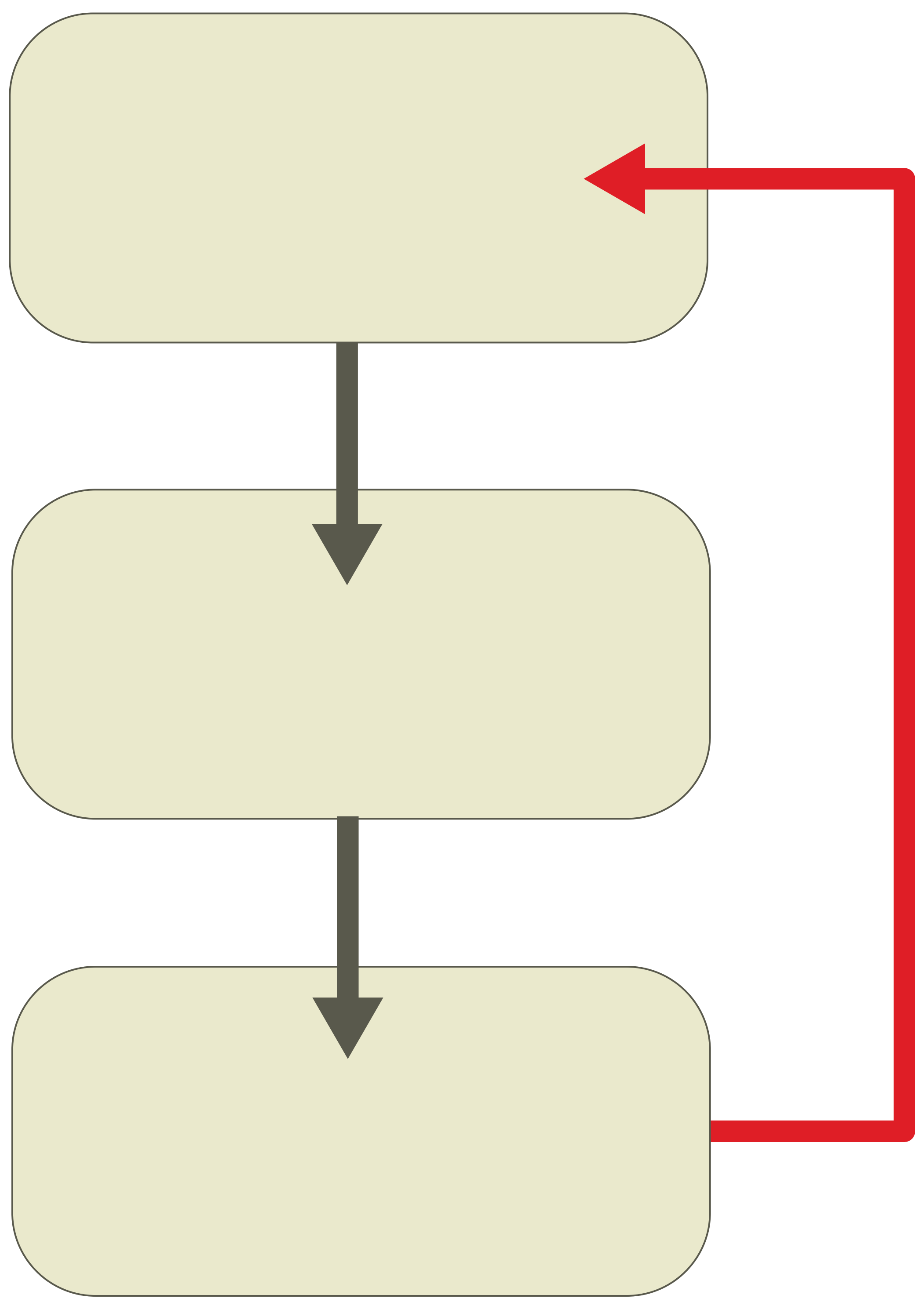
This presentation will focus on Custom Models.

What are Custom Models



Custom Models enable Xandr’s UI, API, and Data Science Toolset users to add decision making logic for Bid Valuation, Creative Delivery and Non-Valuation purposes to their Augmented Line Items (ALI) or campaigns.

How Custom Models are used

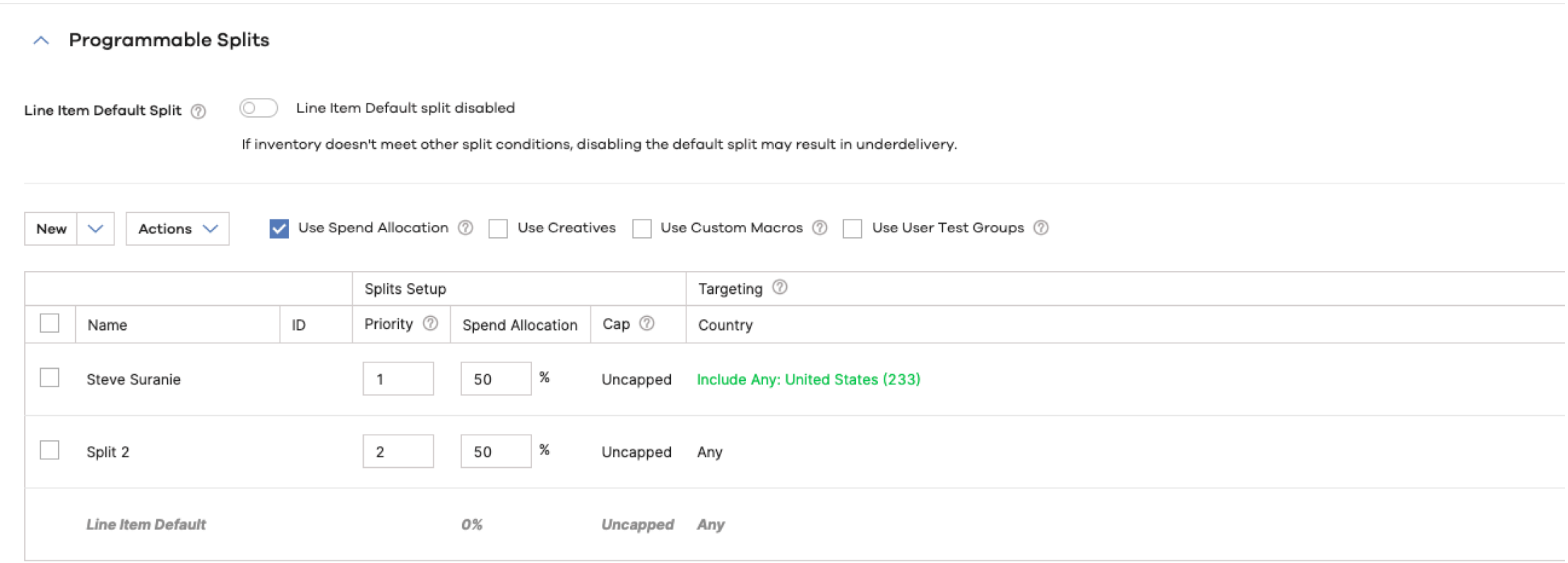


Custom Models can be created by:

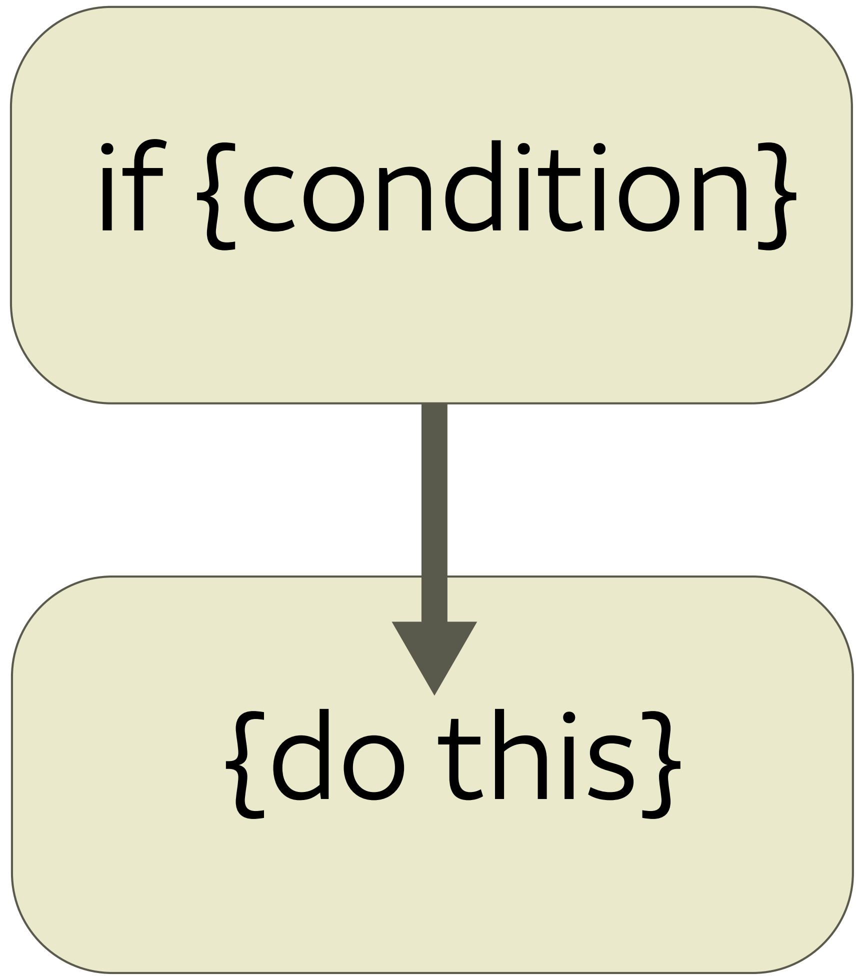
- Using the UI to create a split on the ALI.
- Using Xandr's API's Split Service.
- Creating a decision tree using the Bonsai language.
- Creating a logistic regression model.

Splits in Invest

Users can create splits on Line Items by logging into Invest, opening a Line Item and going to the Programmable Splits section.



Split Service



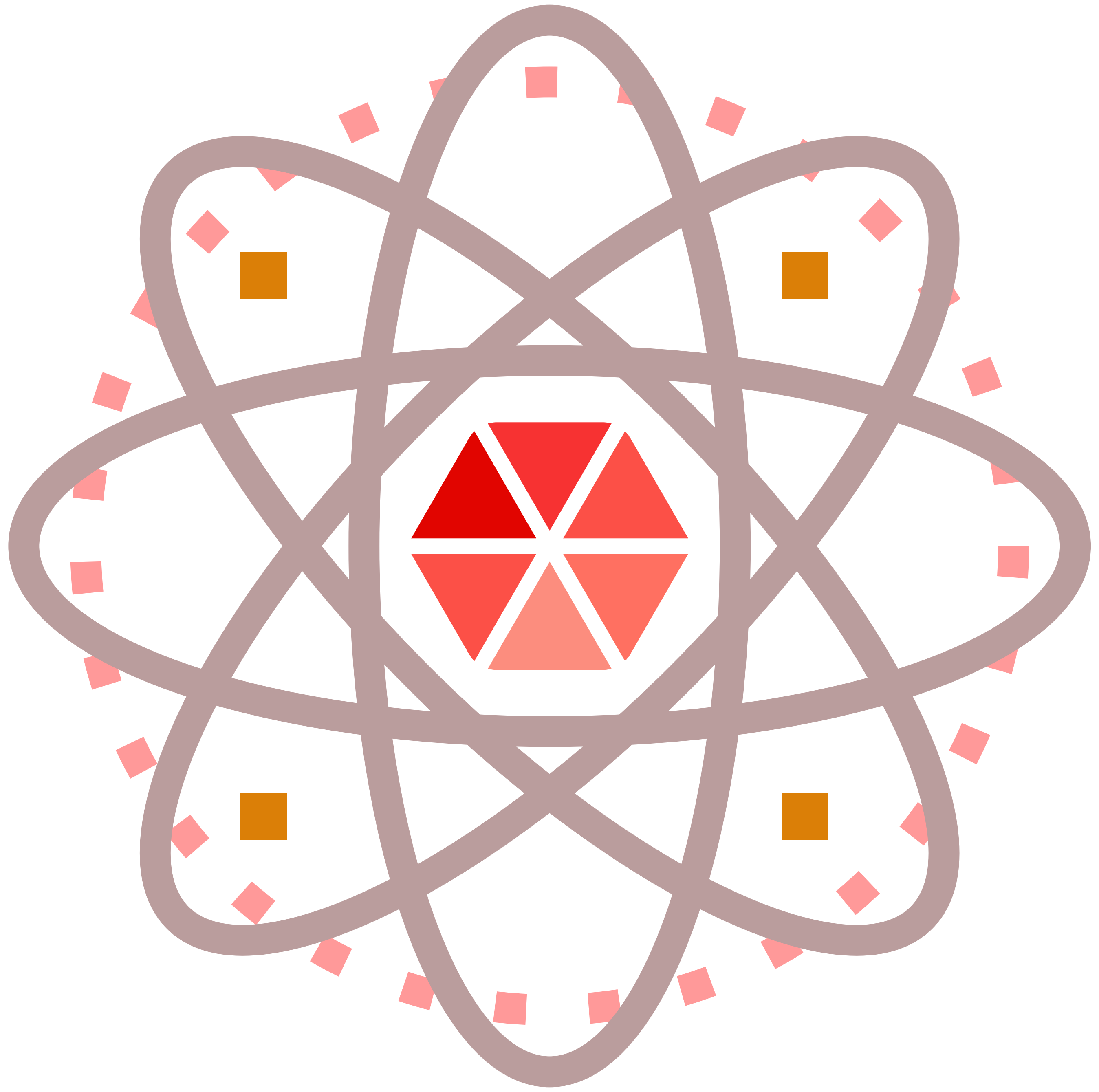

```
…"bid_modifier": 0.5,
    ”conditions”: [
        {
            "field": "browser",
            "operator": "in",
            "value": [
                8
            ]
        }
    ], …
```

Users can also utilize the Xandr API split service. This service enables users to create, update and delete splits on an ALI by passing a JSON file with the split data to the Xandr API.

The same process occurs in the background of the UI. When submitted, the data is collected and an API call is made for the splits.

Both the UI and Split Service have a limit of 100 conditions.

Why Use The Data Science Toolkit (DST)



DST provides tools for users to expand Custom Models beyond the 100 condition limit and to incorporate probability and predictability on end user actions.

DST provides two options:

- Bonsai decision trees
- Logistic regression models

Bonsai Decision Trees

Bonsai is a proprietary computing language developed by Xandr. Its simplicity enables non-programmers to write conditions and the code can be outputted from scripts.

A decision tree written in Bonsai is structured as a branch or a series of branches written as if/elif/else and/or switch expressions. The outcome of a branch is known as a leaf or leaf value.

Bonsai Decision Trees

A Bonsai if/elif/else example:

```
if country = "US":
  1
elif user_hour range (8, 12):
  if domain = "news.com":
    0.85
  else:
    0.2
else:
  0.1
```

The decision path of this example is as follows:

1. Is the user located in the U.S.?
   a. If yes, bid 1
   b. If no, go to step 2
2. Is it between 8:00AM and 12:00PM in the user's time zone?
   a. If yes, go to step 3
   b. If no, bid 0.1
3. Are they viewing news.com?
   a. If yes, bid 0.85
   b. If no, bid 0.2

Bonsai Decision Trees

A Bonsai switch example:

```
switch user_hour:
    case (1 .. 3):
        0.4
    case (4, 6, 8):
        if country present:
            0.8
        else:
            0.4
    default: 0.1
```

The decision path of this example is as follows:

1. Is the user’s timezone between 1:00 and 3:00 AM?
    a. If yes, bid 0.4
    b. If no, go to step 2
2. Is it 4, 6 or 8 AM in the user’s time zone?
    a. If yes, go to step 3
    b. If no, bid 0.1
3. Can we determine the user’s country?
    a. If yes, bid 0.8
    b. If no, bid 0.4

Bonsai trees = Splits

Bonsai trees and Splits are similar in that they:

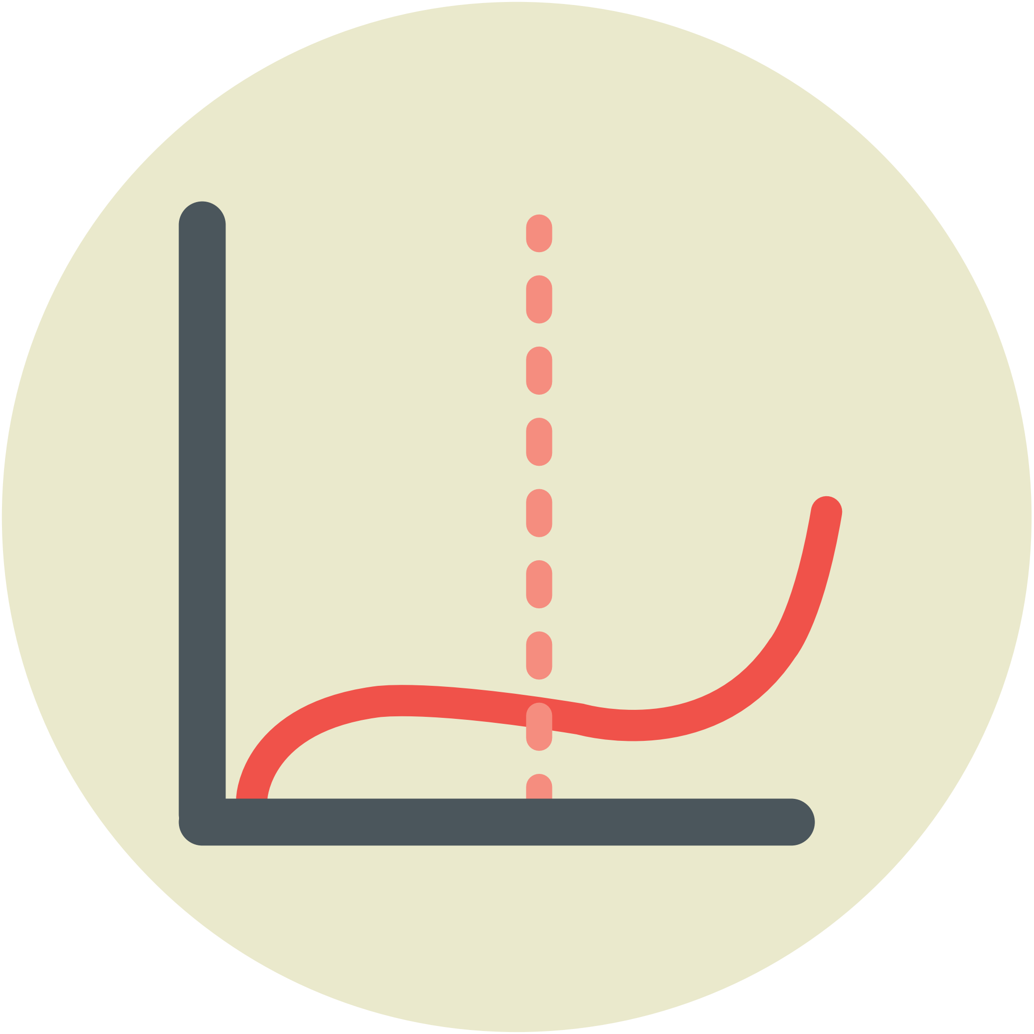
- Are attached to an ALI.
- Programmatically change line item configurations based on conditional input.

Bonsai trees ≠ Splits

Bonsai trees and Splits differ:

- Splits have limited conditions. Bonsai trees do not and can have nested conditions.
- There can only be 100 splits associated with an ALI.* Bonsai trees can have thousand of branches and leaves.
- Bonsai trees require some basic coding skills.

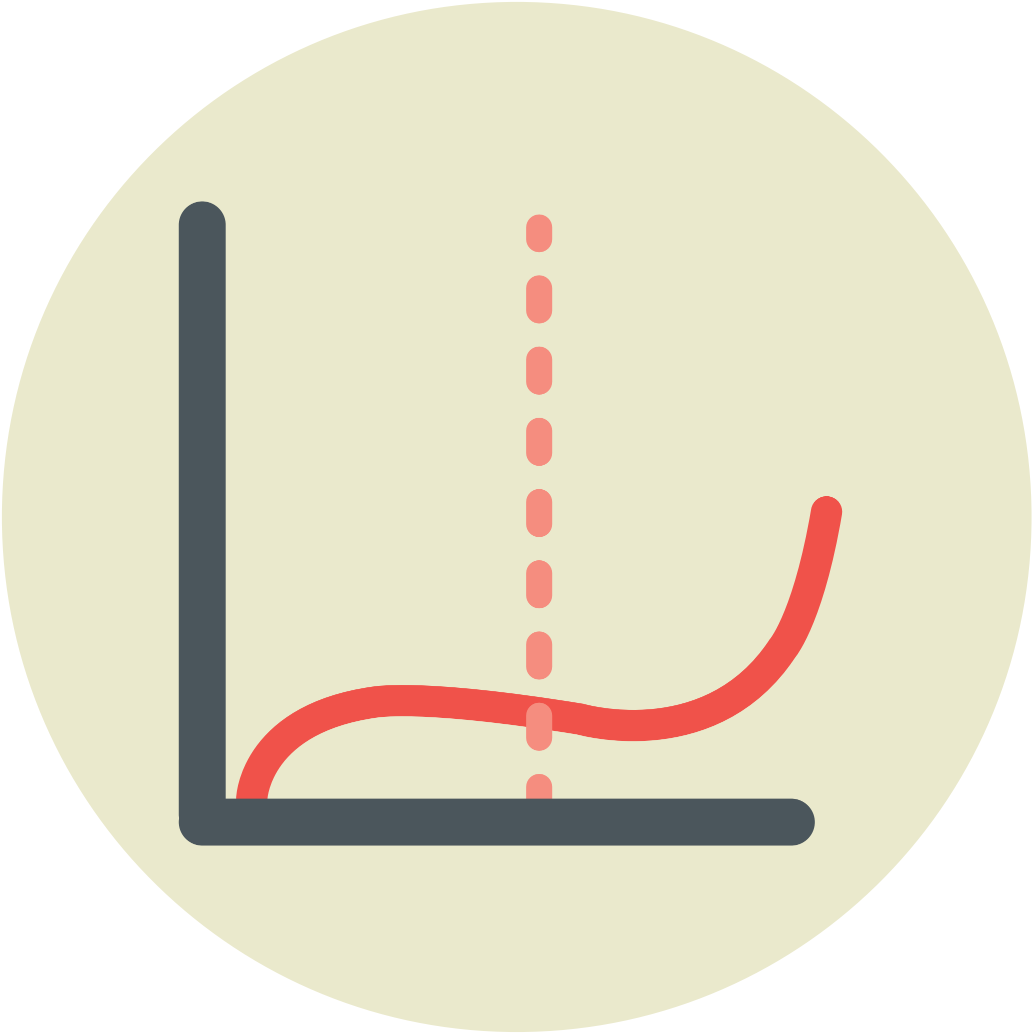
Logistic Regression Models



Logistic Regression models are the data science part of DST. A logistic regression model is the basic approach to predict the probability of a binary response (click or don't click; buy or don't buy) from a combination of multiple signals.

By utilizing logistic regression data scientists can run expressive models that produce more accurate predictions and that can quickly be trained at a high scale. By building tailored algorithms, clients with sophisticated data science tools can achieve better performance than our built-in optimization and can run complex offline models in real time.

Logistic Regression Models



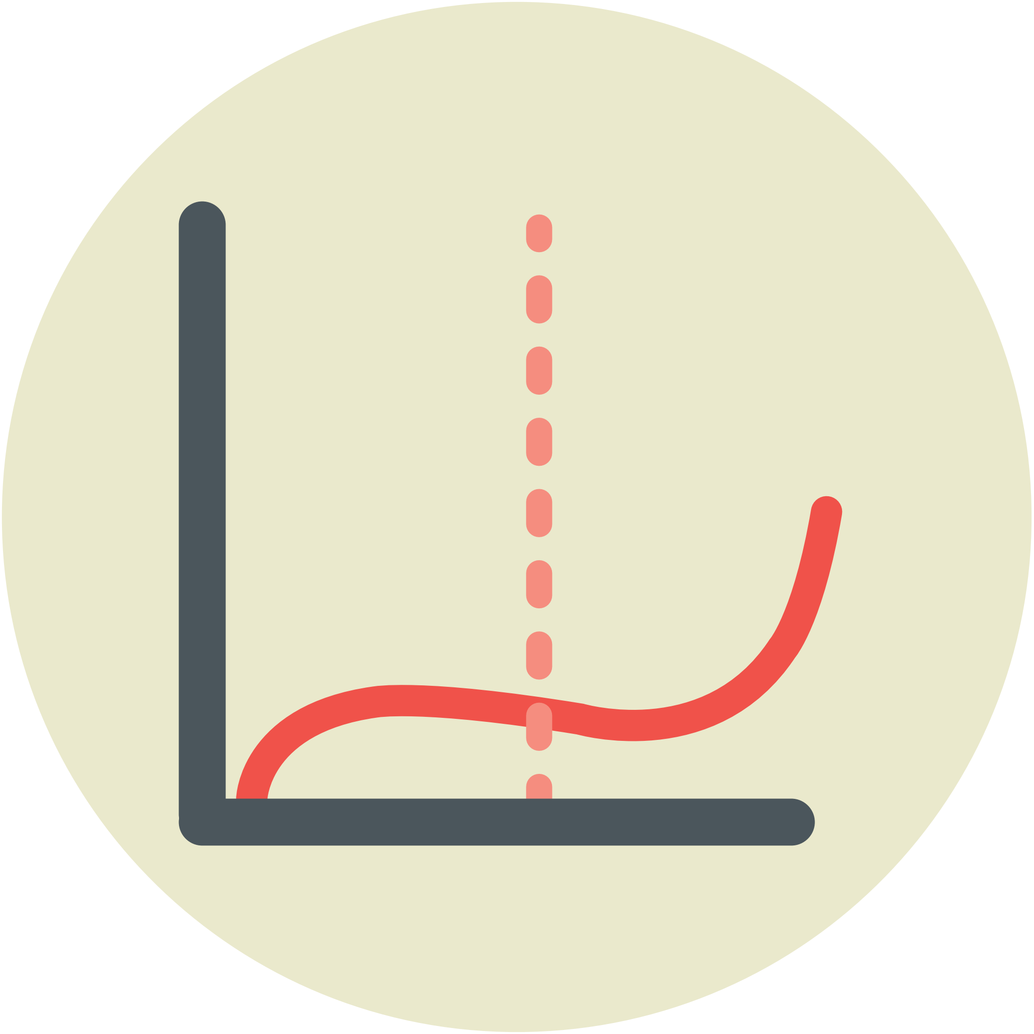
The formula for Logistic Regression starts with determining the probability of an event (user click, a pixel fired, etc.)

$$p = \mathrm{Prob}\,(\mathrm{event} = 1 |\, x_1, \ldots, x_n) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \ldots \beta_n x_n))}$$

The probability (p) the above model is looking to prove is whether an event will occur (event = 1), if we want to prove an event would not occur the event would equal 0.

The probability is also conditional on predictors (χ_1 etc), variables that represent the features in a bid request and beta coefficients (β), weights that the model applies to the predictors.

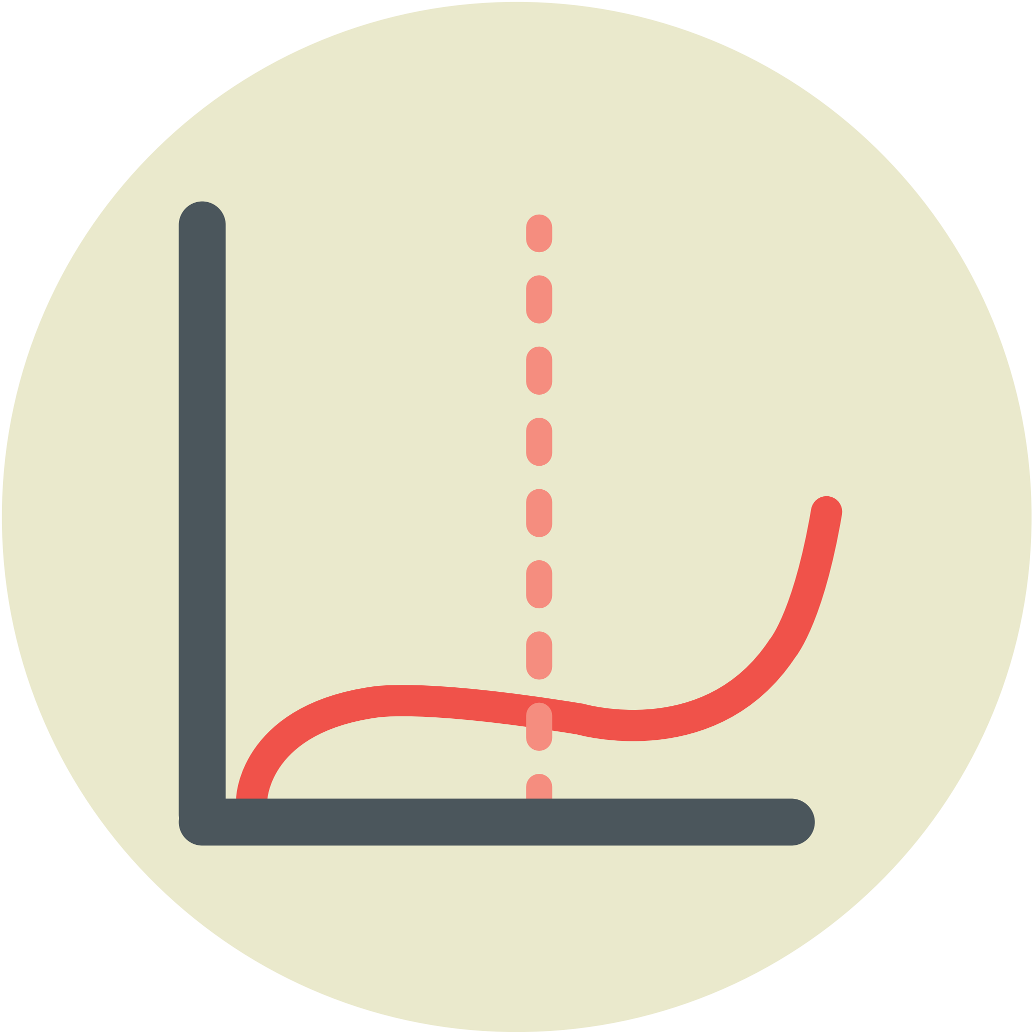
Logistic Regression Models



Categorical Predictors (χ_n):
Ad bidding has a lot of categorical predictors, such as browser, domain (cnn.com, msnbc.com, etc.), day of the week and so on.

These are assigned a value of 1 or 0 (this process is called one-hot encoding). So let's say we are using browsers as a categorical predictor, χ_1 would be 1 if "browser = safari" and 0 if not, χ_2 would be 1 if "browser = firefox", and so on…

Logistic Regression Models



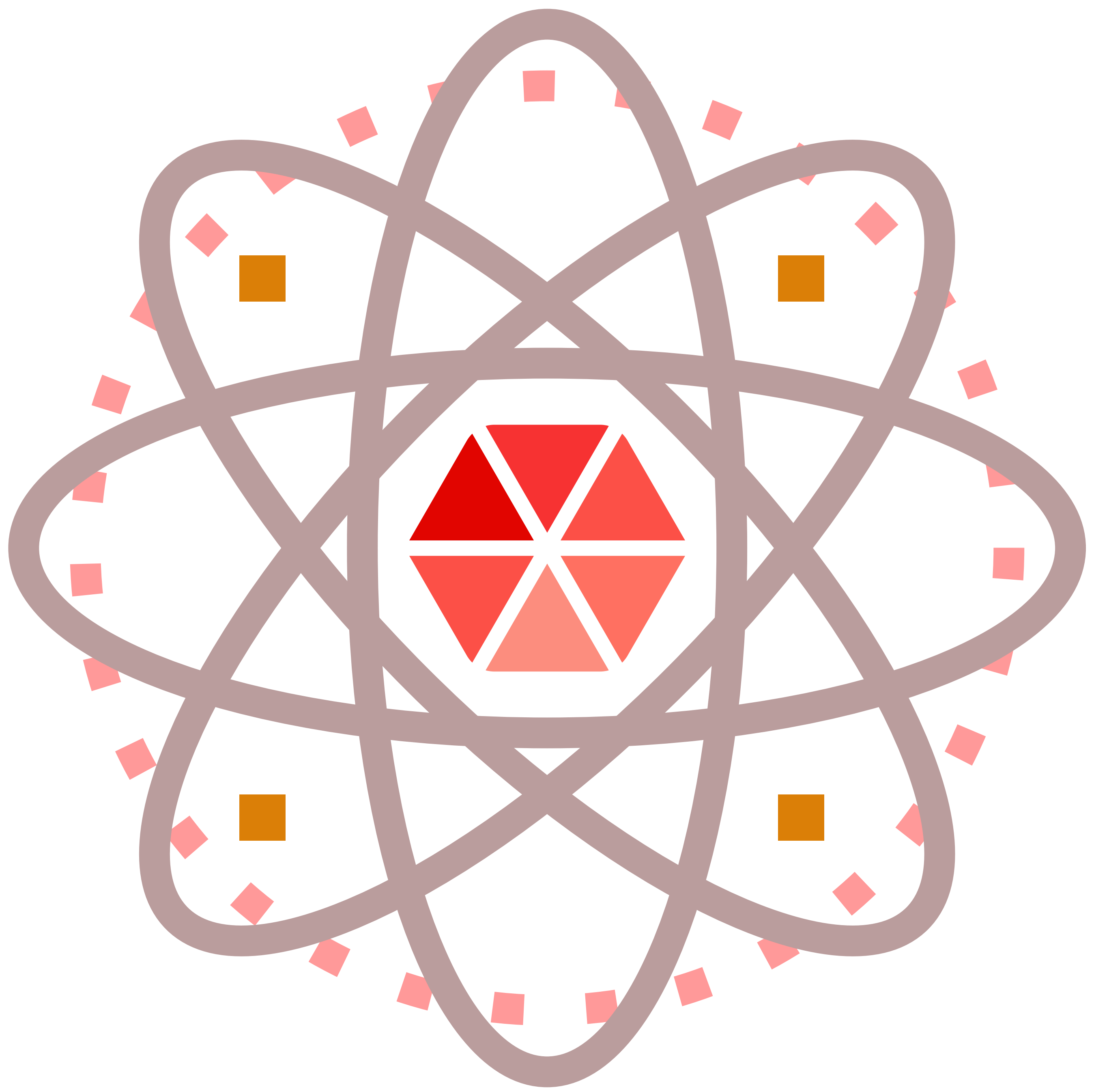
Beta Coefficients (β)
The beta coefficients are the weights that the model assigns to the different predictors.

Browser	Coefficient
Safari	1.2
Firefox	0.8

So if χ_1, is Safari then the value would be 1x1.2

$$p = \frac{1}{1 + \exp(-(\beta_0 + 1.2x_{\text{safari}} + 0.8x_{\text{firefox}} + \cdots + \beta_n x_n))}$$

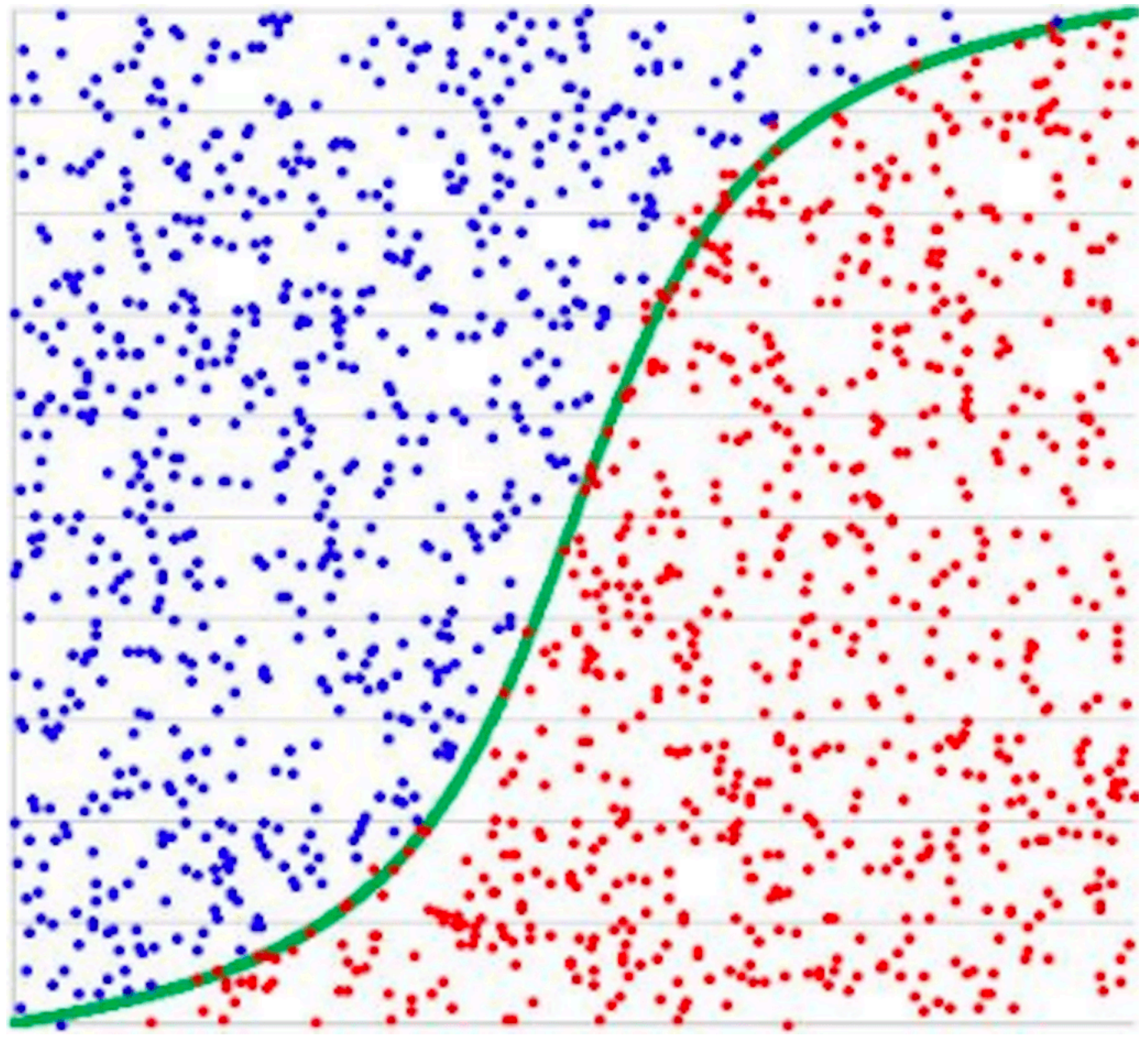
Logistic Regression Models



Other factors:
There are other factors used in determining probability which increases the complexity of the equation but also improves the accuracy of the answer, such as one-hot encoding, higher-order predictors, hashed predictors, weight vectors, etc.

$$p = \frac{1}{1 + \exp(-(\beta_0 + 1.1x_{\text{safari:cnn.com}} + 1.3x_{\text{safari:nytimes.com}} + 1.2x_{\text{safari:yahoo.com}} + 3.3x_{\text{firefox:cnn.com}} + 0.7x_{\text{firefox:nytimes.com}} + 0.1x_{\text{firefox:yahoo.com}} + \cdots + \beta_n x_n))}$$

Translating Probability to Goal Value



The probability of an event happening is then used to determine the expected value (v) of that event, for example, the effective cost per click (eCPC). This is determined by multiplying the probability by the goal value.

The formula for deriving an expected value for an impression from the probability of an event happening is:

$$v_0 = \text{goal\_value} \cdot p = \frac{\text{goal\_value}}{1 + \exp(-(\beta_0 + \beta_1 x_1 + \ldots \beta_n x_n))}$$

What Happens At Auction Time

Once the line item passes targeting, Xandr uses the line item's logistic regression model to determine a bid price.
This is a simplified explanation.

1. Xandr determines if there are categorical predictors in the bid that the logistic regression model requires.
2. Xandr checks for attached Bonsai trees, parses their features and adds them to the model.
3. Xandr then sums the components and passes that to the logistic function to compute the estimated probability of a click.
4. Xandr then uses the expected value and the amount of inventory available to compute a bid.