

Data Science Toolkit: ALI Workflow With Custom Models

Working with [Custom Models](#) such as [Bonsai trees](#) or Logistic Regression models and Augmented Line Items (ALI) is fairly similar to the process of working with traditional campaigns. This guide provides instructions for creating and associating custom models to an ALI. In the example below, we will create a Bonsai expected value [decision tree object](#) and use it as a bidding strategy for an ALI.

Tree creation and validation

The syntax for Bonsai trees and the features currently available for campaigns are the same for trees that work with ALI. Use the Custom Model Service to create your custom model. The Custom Model Parser Service can be utilized to validate the Bonsai decision tree.

Documentation for these services can be found [here](#):

- [Custom Model Service](#)
- [Custom Model Parser Service](#)

Example

For this example, we will create a simple tree under a test advertiser:

Custom model

We will use the JSON below to create the custom_model and pass it to the API using POST and a curl command.

```
1 {
2   "custom_model": {
3     "name": "Test Custom Model",
4     "code": "mcg-test-custom-model",
5     "advertiser_id": 354236,
6     "custom_model_structure": "decision_tree",
7     "model_output": "bid",
8     "model_text": "MAo="
9   } }
```

```
1 $ curl -b dcc -c dcc -X POST -s -d '@json/custom-model.json'
2 "https://api-test.appnexus.com/custom-model?advertiser_id=354236"
```

If the request is successful a JSON response will be returned. The response in this example is abbreviated.

```
1 {
2   "response": {
3     "status": "OK",
4     "count": 1,
5     "id": "379792",
6     "start_element": 0,
7     "num_elements": 100,
8     "custom_model": {
9       "id": 379792,
10      "name": "Test Custom Model",
11      "code": "mcg-test-custom-model",
12      "member_id": 958,
13      "advertiser_id": 354236,
14      "custom_model_structure": "decision_tree",
15      "model_output": "bid",
16      "compiled_text": null,
```

```

17     "model_text": "MAo=",
18     "original_text": "0",
19     "active": true,
20     "last_modified": "2016-08-30 20:55:38"
21   },
22   "dbg_info": {...}
23 }
24 }

```

Creating an ALI through the API

Custom models enable Xandr API users to add decision-making logic to the valuation models, non-valuation models, and creative delivery of their Augmented Line Items or campaigns. For instructions on creating an ALI, [review our ALI Creation documentation](#).

Associating a custom model with an ALI

On each augmented line item there is an array field titled `custom_models` which lists custom models associated with the ALI. Instructions for associating a custom model with an ALI can be found in the [Line Item Model Service documentation](#).

Targeting IP addresses

Customers in countries that conform to GDPR should be aware that when targeting IP addresses or IP ranges, either in your line item, Bonsai tree, or logistic regression model, there is a possibility the address may be truncated. For details on this process, see our [Service Policies](#).

Multiple custom models


You will be able to associate multiple custom models of different types to the augmented line item. These include several custom model types that can be used to override portions of the impression's expected value calculation. An example below demonstrates how the process of associating multiple custom models to a Line Item may work:

Custom model types

The following model types can be associated as custom models to your Line Item:

Model Name	Model Output	Output Type / Range
<code>expected_value</code>	Overrides the Xandr optimized expected value of the impression.	Float
<code>creative_selection</code>	Dynamic selection of creatives associated to this line item.	Creatives, for more information, see Creative Selection Custom Model .
<code>ev_click</code>	Expected value of a click for this impression.	Float
<code>click_imp</code>	Probability of a click for this impression.	Float - (0,1)
<code>ev_conv</code>	Expected value of a conversion for this impression.	Float
<code>conv_imp</code>	Probability of a conversion for this impression.	Float - (0,1)

<code>conv_click</code>	Probability of a conversion, given a click.	Float - (0,1)
<code>bid_modifier</code>	Amount by which this bid on this impression will be multiplied.	Float
<code>nonvaluation</code>	Custom Macros, Learn Status, and other nonvaluation nodes.	Custom model output does not affect bid, though it will be used in reporting when the feature is implemented. For more information, see Non- Valuation Custom Model .
<code>cadence</code>	Cadence modifier for this impression.	Float

 A 'goal_type' of 'custom' on your Line Item is only required in some scenarios, such as associating an 'expected_value' model with your Line Item. Certain optimization features, such as Ranked Discovery will be disabled when using this goal_type with your custom models.

Read the Custom Models page to learn how the bid calculation is affected by Line Item goals and associated component models.