Routes (follows from base URL) BF is for Basic Flow

BF - 2

- /scenarios - Returns a JSON object literal of all scenarios to report on (currently, there just exists Base Case, Scenario A, and Scenario B).

```
{ SCENARIOS:
        [
                {Scenario Schema JSON object literals}
        ]
}
```

BF - 4

- /scenarios/:id - Returns a JSON object literal of a specific scenario by an ID to report on (currently, there just exists Base Case, Scenario A, and Scenario B).

```
{Scenario Schema JSON object literals}
```

BF - 5 to 7

- /scenarios/:id/nodes/:Group?sort=DAY&metric=LMP&id=(Whatever lol)

How this works:

Group: Indicates how the node data is being clustered. By name, region, etc...
Right now, we only have names. Default: None.
Sort: Indicates how we will group the data. All, yearly, etc...
Should be able to support daily, monthly, quarterly, yearly, and all. Default: daily.
Metric: Indicates what metric we will report on.
Right now, should just report on LMP. Default: LMP.
Id: A specific ID associated with the clustering method, returns information just on that cluster.
Ex: Group = PNODE_NAME, id = 'KENT' will report for nodes for the KENT node.
Id does nothing if no Group is specified (Cause all data is treated as one set)

What it will return:

Nested JSON object literals in this format:

```
{
        NodeGrouping (Ex: If group = PNODE_NAME, NodeGrouping = UN.ALTA 345 UALT)
        {
        TimeGrouping (Ex: If sort = QUARTER, TimeGrouping = 2022-Q1 )
        {
                        MEAN: Average of the chosen metric for all nodes in this cluster.
                        MEDIAN: Median of the chosen metric for all nodes in this cluster.
                        STD: Standard Deviation of the chosen metric for all nodes in this cluster.
        }
        }
}
```

Examples:

- User wants to know summary statistics of the locational marginal price (LMP) at a particular node (KENT) by calendar quarter for a specific scenario (1).

Request to
/scenarios/1/nodes/PNODE_NAME?sort=QUARTER&metric=LMP&id=KENT

- User wants to know the average LMP of all nodes over monthly indexes (for the heatmap) for comparing between scenario 1 and 2.

Request to
/scenarios/1/nodes?sort=MONTH&metric=LMP
/scenarios/2/nodes?sort=MONTH&metric=LMP

Decide later if we want backend to be able to perform more calculations so that frontend just has to display data, or let frontend take care of calculating maximum absolute percent errors

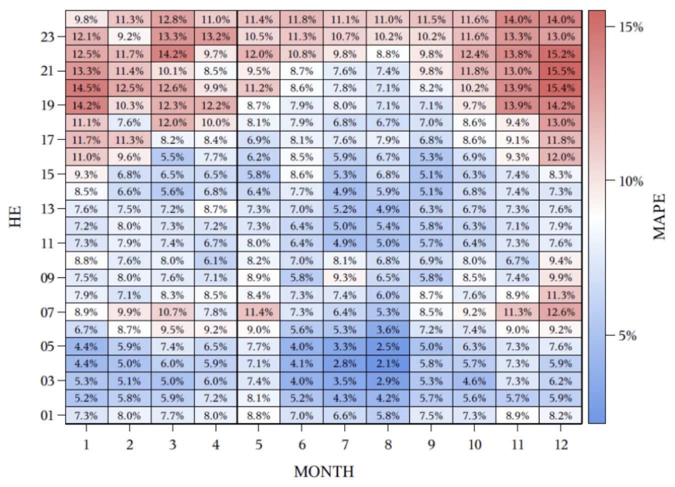
<u>//This is deprecated</u> BF - 5 to 7

- /scenarios/:id/nodes - Returns a JSON object of all nodes associated with the specific scenario by an ID.

  Allows for additional query parameters for:
  Metric - Specify a specific metric to report on (Currently just LMP) **BF - 5**
  Timing - Daily, Monthly, Quarterly, Yearly, All (RN it’s one day) **BF - 6**

Name - Group by specific name or ID or hour (for use case 1) **BF - 7**

Graphs:

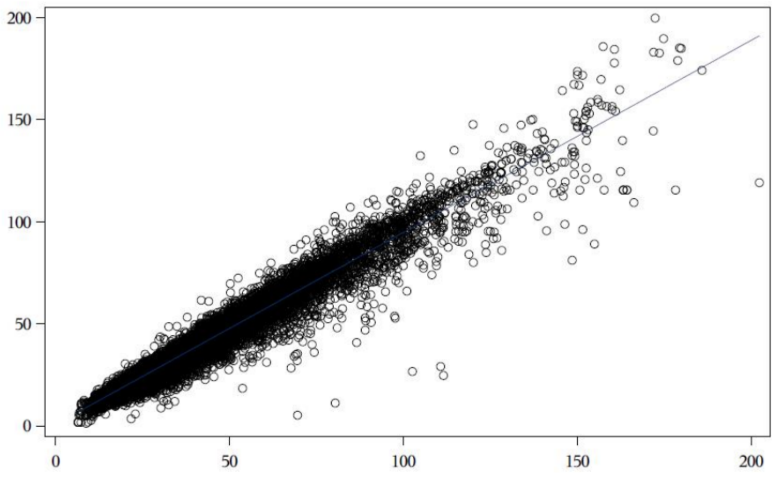


Sample Graphs for Use Case 1

September 29, 2022

Scatter Plot

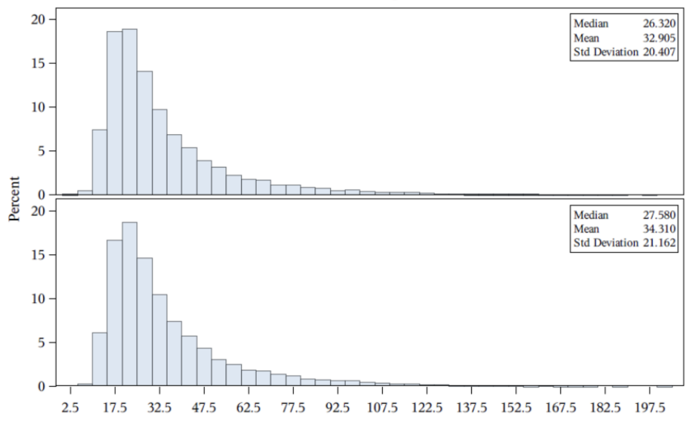
1. Pick a single node.
2. Plot a one point for each hour, where the points X and Y values are:
   - X = LMP from base case
   - Y = LMP from scenario



Histogram

Plot one histogram of the HUB node prices for the base case.

Using the same bins, plot a second histogram of the HUB not prices for scenario you are validating.



Maximum Absolute Percent Errors

$$\text{The maximum absolute percent error} = \frac{LMP_{BASE} - LMP_{SCENARIO}}{LMP_{BASE}}$$

One value is computed for each hour ending and month combination in this example.