

How to Use Lookup Functions to Simplify Model Setups in hēRo3

February 2021

How to Use Lookup Functions in hēRo3: What is *look_up*, and When Might You Want To Use It?

- ***look_up(...)*** is convenience function in hēRo3 that lets you pull values from a table into a model:
 - It is similar to ***vlookup(...)*** in Excel
- Reasons why you might want to use ***look_up(...)*** instead of hardcoding values in a model:
 - All values for an input variable can be conveniently found in a single place (ie, a hēRo3 table)
 - Models can be built much faster when all values for a variable are populated using ***look_up(...)*** and a corresponding hēRo3 table
 - If you use ***look_up(...)*** to populate values for a variable, new or updated input values can be easily pulled into a model simply by copying and pasting them into an existing hēRo3 table

How to Use Lookup Functions in hēRo3: What is *look_up*, and When Might You Want To Use It? (Cont.)

- Common use cases of *look_up(...)* include:
 - Patient characteristics by subgroup
 - Monitoring costs by health-state
 - Probability of death by age
 - Health-state disutilities by adverse event
- More complex use cases can involve multiple arguments in a function, such as determining health-state utility values dependent on group and health-state

How to Use Lookup Functions in hēRo3:

look_up Syntax & Arguments

- Syntax:
 - *look_up*(data, . . ., value = “value”, bin = FALSE)
- Arguments:
 - **data** – hēRo3 table containing values
 - . . ., – Individual characteristics, corresponding to columns in data table
 - **value** – Value to extract from table
 - **bin** [optional] – TRUE = approximate match; FALSE = exact match (default)
 - While default for *vlookup*(...) in Excel is to do an approximate match, default for *look_up*(...) in hēRo3 is to do an exact match (ie, bin = FALSE)

How to Use Lookup Functions in hēRo3: Looking Up A Single Numeric Value

- Example below looks up probability of survival from table, *km_table*, for particular value of time (ie, 0.2)

```
fx | look_up(km_table, time = 0.2, value = "survival", bin = FALSE)
```



How to Use Lookup Functions in hēRo3: Looking Up A Single Numeric Value (Cont.)

- Table called ***km_table*** contains survival probabilities for different values of variable, **time**

```
fx | look_up(km_table, time = 0.2, value = "survival", bin = FALSE)
```

Table called, ***km_table***

	A	B
1	time	survival
2	0	1
3	0.2	0.95
4	0.4	0.89
5	1.2	0.86
6	2.6	0.8
7	5.4	0.76
8	7.8	0.72
9	10.2	0.65
10	16.4	0.6
11	20.1	0.58

How to Use Lookup Functions in hēRo3: Looking Up A Single Numeric Value (Cont.)

- Value of survival probability is needed when **time** = 0.2

```
fx | look_up(km_table, time = 0.2, value = "survival", bin = FALSE) |
```

Table called, *km_table*

	A	B
1	time	survival
2	0	1
3	0.2	0.95
4	0.4	0.89
5	1.2	0.86
6	2.6	0.8
7	5.4	0.76
8	7.8	0.72
9	10.2	0.65
10	16.4	0.6
11	20.1	0.58

Argument 1 has function
look for value = 0.2
in column, time

How to Use Lookup Functions in hēRo3: Looking Up A Single Numeric Value (Cont.)

- Survival probabilities are contained in column, **survival**

```
fx | look_up(km_table, time = 0.2, value = "survival", bin = FALSE) |
```

Table called, *km_table*

	A	B
1	time	survival
2	0	1
3	0.2	0.95
4	0.4	0.89
5	1.2	0.86
6	2.6	0.8
7	5.4	0.76
8	7.8	0.72
9	10.2	0.65
10	16.4	0.6
11	20.1	0.58

Value from column, **survival**, is returned for time = 0.2

How to Use Lookup Functions in hēRo3: Looking Up A Single Numeric Value (Cont.)

- Since exact match is desired, bin is set to FALSE, and value of 0.95 is returned by *look_up(...)*

```
fx look_up(km_table, time = 0.2, value = "survival" bin = FALSE)
```

Table called, *km_table*

	A	B
1	time	survival
2	0	1
3	0.2	0.95
4	0.4	0.89
5	1.2	0.86
6	2.6	0.8
7	5.4	0.76
8	7.8	0.72
9	10.2	0.65
10	16.4	0.6
11	20.1	0.58

Match must
be exact

How to Use Lookup Functions in hēRo3: Looking Up A Set of Values

- Example below looks up mortality probability in relation to ***current_age*** from table called ***mortality***

```
fx | look_up(mortality, "Age" = current_age, value = "Prob", bin = TRUE) × ✓
```

How to Use Lookup Functions in hēRo3: Looking Up A Set of Values (Cont.)

- Table called *mortality* contains annual probabilities of death by chronological age

```
fx | look_up(mortality, "Age" = current_age, value = "Prob", bin = TRUE) |
```

Table called, *mortality*

	A	B
1	Age	Prob
2	30	0.0209
3	31	0.02474
4	32	0.02891
5	33	0.03344
6	34	0.03822
7	35	0.04319

How to Use Lookup Functions in hēRo3: Looking Up A Set of Values (Cont.)

- Value of mortality is needed when **Age** = *current_age* (in years)

```
fx | look_up(mortality, "Age" = current_age, value = "Prob", bin = TRUE) |
```

Table called, *mortality*

	A	B
1	Age	Prob
2	30	0.0209
3	31	0.02474
4	32	0.02891
5	33	0.03344
6	34	0.03822
7	35	0.04319

Argument 1 looks for *current_age* in column, **Age**, where *current_age* equals

```
start_age + model_year - cycle_length_years
```

How to Use Lookup Functions in hēRo3: Looking Up A Set of Values (Cont.)

- Mortality probabilities by chronological age are contained in column, **Prob**

```
fx | look_up(mortality, "Age" = current_age, value = "Prob", bin = TRUE) |
```

Table called, *mortality*

	A	B
1	Age	Prob
2	30	0.0209
3	31	0.02474
4	32	0.02891
5	33	0.03344
6	34	0.03822
7	35	0.04319

Column from which
value is returned for
mortality probability

How to Use Lookup Functions in hēRo3: Looking Up A Set of Values (Cont.)

- All values in column, **Prob**, are returned, since bin = TRUE

```
fx look_up(mortality, "Age" = current_age, value = "Prob", bin = TRUE)
```

Table called, *mortality*

	A	B
1	Age	Prob
2	30	0.0209
3	31	0.02474
4	32	0.02891
5	33	0.03344
6	34	0.03822
7	35	0.04319

Does not require an exact match if *current_age* is not an integer

How to Use Lookup Functions in hēRo3: Looking Up Values with An Indicator Variable

- Example below looks up value for an efficacy measure, **prb_EASI_90_Mod**, from table called **Scenarios**, using variable, **Scenario_Flag**

```
fx | look_up(Scenarios, Variable = "prb_EASI_90_Mod", value = Scenario_Flag) |
```



How to Use Lookup Functions in hēRo3: Looking Up Values with An Indicator Variable (Cont.)

- Table, **Scenarios**, contains probabilities of attaining different response thresholds for **Low**, **Basecase**, and **High** efficacy scenarios

```
fx look_up(Scenarios, Variable = "prb_EASI_90_Mod", value = Scenario_Flag)
```

Table called, *Scenarios*

	A	B	C	D
1	Variable	Low	Basecase	High
2	prb_EASI_50_Mod	0.10	0.13	0.15
3	prb_EASI_75_Mod	0.20	0.23	0.25
4	prb_EASI_90_Mod	0.50	0.55	0.60
5	prb_EASI_50_Sev	0.20	0.25	0.30
6	prb_EASI_75_Sev	0.09	0.1	0.11
7	prb_EASI_90_Sev	0.50	0.53	0.55

How to Use Lookup Functions in hēRo3: Looking Up Values with An Indicator Variable (Cont.)

- In present instance, we want probability of meeting response threshold when **Variable = prb_EASI_90_mod**

fx look_up(Scenarios Variable = "prb_EASI_90_Mod" value = Scenario_Flag) X ✓

Table called, *Scenarios*

	A	B	C	D
1	Variable	Low	Basecase	High
2	prb_EASI_50_Mod	0.10	0.13	0.15
3	prb_EASI_75_Mod	0.20	0.23	0.25
4	prb_EASI_90_Mod	0.50	0.55	0.60
5	prb_EASI_50_Sev	0.20	0.25	0.30
6	prb_EASI_75_Sev	0.09	0.1	0.11
7	prb_EASI_90_Sev	0.50	0.53	0.55

Argument 1 has function look for value, **prb_EASI_90_mod**, in column, **Variable**

How to Use Lookup Functions in hēRo3: Looking Up Values with An Indicator Variable (Cont.)

- Probabilities of meeting response thresholds for different values of ***Scenario_Flag*** are contained in columns, **Low**, **Basecase**, and **High**

```
fx look_up(Scenarios, Variable = "prb_EASI_90_Mod", value = Scenario_Flag)
```

Table called, *Scenarios*

	A	B	C	D
1	Variable	Low	Basecase	High
2	prb_EASI_50_Mod	0.10	0.13	0.15
3	prb_EASI_75_Mod	0.20	0.23	0.25
4	prb_EASI_90_Mod	0.50	0.55	0.60
5	prb_EASI_50_Sev	0.20	0.25	0.30
6	prb_EASI_75_Sev	0.09	0.1	0.11
7	prb_EASI_90_Sev	0.50	0.53	0.55

Argument 2 returns value from column based on *Scenario_Flag*, which can be set to “Low”, “Basecase”, or “High” in this example

How to Use Lookup Functions in hēRo3: Looking Up Values Using More Than One Argument

- Example below looks up health-state utility values from table, *utility*

```
fx | look_up(utility, group_name = group, state = "healthy", value = "utility")
```

How to Use Lookup Functions in hēRo3: Looking Up Values Using More Than One Argument (Cont.)

- Table, *utility*, contains utility values by health state and group

```
fx | look_up(utility, group_name = group, state = "healthy", value = "utility")
```

Table called, *utility*

	A	B	C
1	group_name	state	utility
2	group_1	healthy	0.77
3	group_1	sick	0.55
4	group_2	healthy	0.88
5	group_2	sick	0.66

How to Use Lookup Functions in hēRo3: Looking Up Values Using More Than One Argument (Cont.)

- Values for variable, **group**, are contained in column, **group_name**

```
fx | look_up(utility, group_name = group, state = "healthy", value = "utility")
```

Table called, *utility*

	A	B	C
1	group_name	state	utility
2	group_1	healthy	0.77
3	group_1	sick	0.55
4	group_2	healthy	0.88
5	group_2	sick	0.66

Argument 1 looks for *group* in column, **group_name**, where **group** is hēRo3 keyword that returns all group names in model.

How to Use Lookup Functions in hēRo3: Looking Up Values Using More Than One Argument (Cont.)

- Correspondingly, values for health state are contained in column, **state**

```
fx | look_up(utility, group_name = group, state = "healthy", value = "utility")
```

Table called, *utility*

	A	B	C
1	group_name	state	utility
2	group_1	healthy	0.77
3	group_1	sick	0.55
4	group_2	healthy	0.88
5	group_2	sick	0.66

Argument 2 looks for "healthy" in column, **state**.

How to Use Lookup Functions in hēRo3: Looking Up Values Using More Than One Argument (Cont.)

- *look_up(...)* returns value from column, **utility**, specific to each health state and group

```
fx | look_up(utility, group_name = group, state = "healthy", value = "utility")
```

Table called, *utility*

	A	B	C
1	group_name	state	utility
2	group_1	healthy	0.77
3	group_1	sick	0.55
4	group_2	healthy	0.88
5	group_2	sick	0.66

Argument 3 returns value from column, **utility**, where both arguments 1 and 2 are met