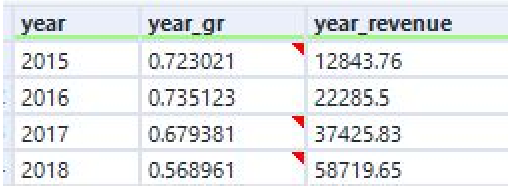
Effective business simulation: revenue stochastic breakdown

(Henry T.H. Tu, 17-nov-2019)

We will break the revenue line item (revenue over years) and cost line item (cost over years) into smaller line items (by region, by channel, by flavour) with stochastic method to avoid monotonic outcome. In the beginning, there are only 4 rows. Then we have 48 rows (4 * 12 months). And we have 240 rows (48 * 5 regions). And we have 1200 rows (240 * 5 channels). And finally we have 6000 rows (1200 * 5 flavours).

3.1. Growth rate and revenue

Let us start with the year revenue and cost as table G1 and we need to break those items down into month. The yearly revenue/cost table can be generated automatically by transforming the year sequence into numbers with the given formulas in table G1.



year	year_gr	year_revenue
2015	0.723021	12843.76
2016	0.735123	22285.5
2017	0.679381	37425.83
2018	0.568961	58719.65

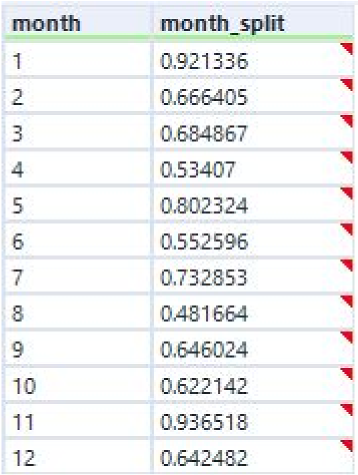
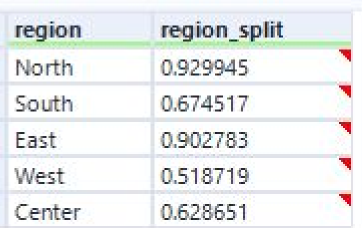
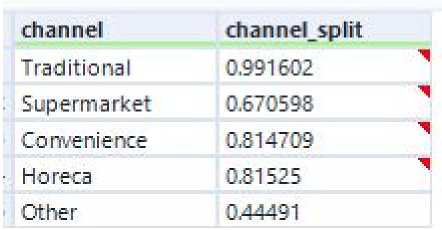
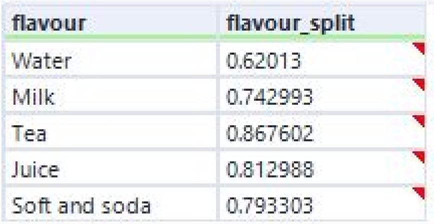
```
------ year
An increasing sequence

------ year_gr
0.5 + 0.25 * tonumber(left(regex_replace(
MD5_ASCII([year] + "/gr")
, "\D+", ""), 7))/pow(10, 7)


------ year_revenue
floor(100 * if year = "2015" then 1243.76
else [Row-1:year_revenue]
* (1 + [year_gr]) endif)/100

------ year_cost
floor(100 * [year_revenue] * (0.8

+ 0.25 * tonumber(left(regex_replace(
MD5_ASCII([year] + "/cost1")
, "\D+", ""), 7))/pow(10, 7)

- 0.25 * tonumber(left(regex_replace(
MD5_ASCII([year] + "/cost2")
, "\D+", ""), 7))/pow(10, 7) )

) / 100
```

Table G1. Revenue and cost by years

3.2. Stochastic breakdown

We try to break the revenue by year into revenue by month with stochastic breakdown process (table G2 and G5) instead of multiplying the month weight with the year revenue to avoid monotonicity. We generate the item_split score, which is the combination of month_split and random fluctuation amount by particular month in year. This amount is to introduce the stochasticity. We need to compute the item_split_Z which is the sum of all items by item_upper. This amount is used to normalize the item_revenue later so that the item_revenue add up to total revenue.

In the beginning, we set item_revenue to the year_revenue. And this is also the starting point for us to break the total into smaller items. Other than that, the stochastic breakdown process is the same. We set the item_upper to the upper group of the item/ We set item_path to the current item level, which is used to generate item_split and item_code later. We sum all item_split with item_upper group to get item_split_Z for each group. Then finally, we multiply the item_revenue with the item_split and divide by item_split_Z to get the breakdown value

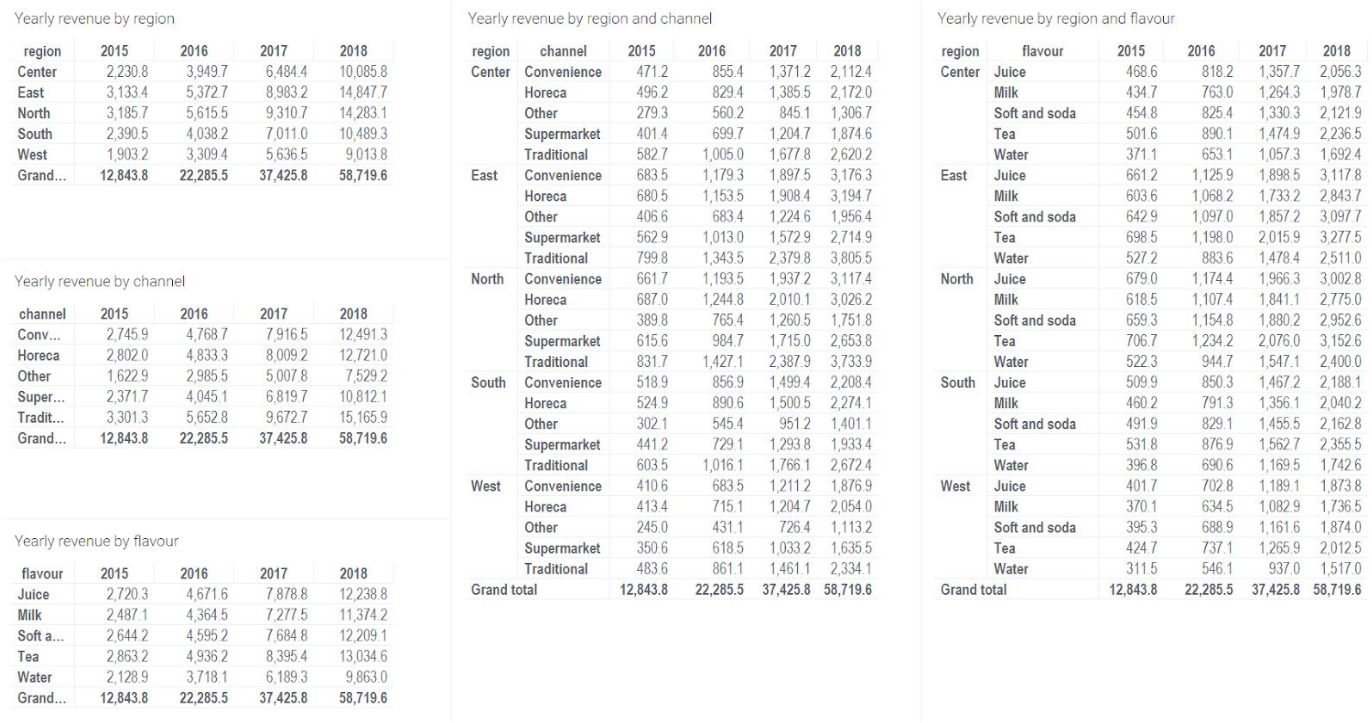
 month | month_split 1 | 0.921336 2 | 0.666405 3 | 0.684867 4 | 0.53407 5 | 0.802324 6 | 0.552596 7 | 0.732853 8 | 0.481664 9 | 0.646024 10 | 0.622142 11 | 0.936518 12 | 0.642482	----- month_split 0.4 + 0.6 * tonumber(left(regex_replace(MD5_ASCII(tostring([month]) + "/month") , "\D+", ""), 7))/pow(10, 7)	----- item_upper [year] ----- item_path [year] + "/" + [month] ----- item_split [month_split] + [User.split_range] * tonumber(left(regex_replace(MD5_ASCII([item_path] + "/split") , "\D+", ""), 7))/pow(10, 7) ----- item_revenue **// [item_revenue] = [year_revenue]** [item_revenue] * [item_split] / [item_split_Z]
 region | region_split North | 0.929945 South | 0.674517 East | 0.902783 West | 0.518719 Center | 0.628651	----- region_split 0.4 + 0.6 * tonumber(left(regex_replace(MD5_ASCII([region] + "/region") , "\D+", ""), 7))/pow(10, 7)	----- item_upper [year] + "/" + [month] ----- item_path [year] + "/" + [month] + "/" + [region] ----- item_split [region_split] + [User.split_range] * tonumber(left(regex_replace(MD5_ASCII([item_path] + "/split") , "\D+", ""), 7))/pow(10, 7) ----- item_revenue [item_revenue] * [item_split] / [item_split_Z]
 channel | channel_split Traditional | 0.991602 Supermarket | 0.670598 Convenience | 0.814709 Horeca | 0.81525 Other | 0.44491	----- channel_split 0.4 + 0.6 * tonumber(left(regex_replace(MD5_ASCII([channel] + "/channel") , "\D+", ""), 7))/pow(10, 7)	----- item_upper [year] + "/" + [month] + "/" + [region] ----- item_path [year] + "/" + [month] + "/" + [region] + "/" + [channel] ----- item_split [channel_split] + [User.split_range] * tonumber(left(regex_replace(MD5_ASCII([item_path] + "/split") , "\D+", ""), 7))/pow(10, 7) ----- item_revenue [item_revenue] * [item_split] / [item_split_Z]
 flavour | flavour_split Water | 0.62013 Milk | 0.742993 Tea | 0.867602 Juice | 0.812988 Soft and soda | 0.793303	----- flavour_split 0.4 + 0.6 * tonumber(left(regex_replace(MD5_ASCII([flavour] + "/flv") , "\D+", ""), 7))/pow(10, 7)	----- item_upper [year] + "/" + [month] + "/" + [region] + "/" + [channel] ----- item_path [year] + "/" + [month] + "/" + [region] + "/" + [channel] + "/" + [flavour] ----- item_split [flavour_split] + [User.split_range]

		* tonumber(left(regex_replace(MD5_ASCII([item_path] + "/split") , "\D+", ""), 7))/pow(10, 7) ----- item_revenue [item_revenue] * [item_split] / [item_split_Z]

Table G2. Stochastic breakdown

3.5. Data exploring

Table G3 show the consistency check and table G4 show the data visualization over historical dimension. If we sum the revenue values by one category or two categories over the years, we must obtain the same numbers in the yearly revenue line item before we perform breakdown. This consistency check is to make sure that there is no error along the way. We can visualize the data with bar chart over years and month. We can also use bar chart or pie chart to see the main drivers (big components) in each category



Yearly revenue by region

region	2015	2016	2017	2018
Center	2,230.8	3,949.7	6,484.4	10,085.8
East	3,133.4	5,372.7	8,983.2	14,847.7
North	3,185.7	5,615.5	9,310.7	14,283.1
South	2,390.5	4,038.2	7,011.0	10,489.3
West	1,903.2	3,309.4	5,636.5	9,013.8
Grand...	**12,843.8**	**22,285.5**	**37,425.8**	**58,719.6**

Yearly revenue by channel

channel	2015	2016	2017	2018
Conv...	2,745.9	4,768.7	7,916.5	12,491.3
Horeca	2,802.0	4,833.3	8,009.2	12,721.0
Other	1,622.9	2,985.5	5,007.8	7,529.2
Super...	2,371.7	4,045.1	6,819.7	10,812.1
Tradit...	3,301.3	5,652.8	9,672.7	15,165.9
Grand...	**12,843.8**	**22,285.5**	**37,425.8**	**58,719.6**

Yearly revenue by flavour

flavour	2015	2016	2017	2018
Juice	2,720.3	4,671.6	7,878.8	12,238.8
Milk	2,487.1	4,364.5	7,277.5	11,374.2
Soft a...	2,644.2	4,595.2	7,684.8	12,209.1
Tea	2,863.2	4,936.2	8,395.4	13,034.6
Water	2,128.9	3,718.1	6,189.3	9,863.0
Grand...	**12,843.8**	**22,285.5**	**37,425.8**	**58,719.6**

Yearly revenue by region and channel

region	channel	2015	2016	2017	2018
Center	Convenience	471.2	855.4	1,371.2	2,112.4
	Horeca	496.2	829.4	1,385.5	2,172.0
	Other	279.3	560.2	845.1	1,306.7
	Supermarket	401.4	699.7	1,204.7	1,874.6
	Traditional	582.7	1,005.0	1,677.8	2,620.2
East	Convenience	683.5	1,179.3	1,897.5	3,176.3
	Horeca	680.5	1,153.5	1,908.4	3,194.7
	Other	406.6	683.4	1,224.6	1,956.4
	Supermarket	562.9	1,013.0	1,572.9	2,714.9
	Traditional	799.8	1,343.5	2,379.8	3,805.5
North	Convenience	661.7	1,193.5	1,937.2	3,117.4
	Horeca	687.0	1,244.8	2,010.1	3,026.2
	Other	389.8	765.4	1,260.5	1,751.8
	Supermarket	615.6	984.7	1,715.0	2,653.8
	Traditional	831.7	1,427.1	2,387.9	3,733.9
South	Convenience	518.9	856.9	1,499.4	2,208.4
	Horeca	524.9	890.6	1,500.5	2,274.1
	Other	302.1	545.4	951.2	1,401.1
	Supermarket	441.2	729.1	1,293.8	1,933.4
	Traditional	603.5	1,016.1	1,766.1	2,672.4
West	Convenience	410.6	683.5	1,211.2	1,876.9
	Horeca	413.4	715.1	1,204.7	2,054.0
	Other	245.0	431.1	726.4	1,113.2
	Supermarket	350.6	618.5	1,033.2	1,635.5
	Traditional	483.6	861.1	1,461.1	2,334.1
Grand total		**12,843.8**	**22,285.5**	**37,425.8**	**58,719.6**

Yearly revenue by region and flavour

region	flavour	2015	2016	2017	2018
Center	Juice	468.6	818.2	1,357.7	2,056.3
	Milk	434.7	763.0	1,264.3	1,978.7
	Soft and soda	454.8	825.4	1,330.3	2,121.9
	Tea	501.6	890.1	1,474.9	2,236.5
	Water	371.1	653.1	1,057.3	1,692.4
East	Juice	661.2	1,125.9	1,898.5	3,117.8
	Milk	603.6	1,068.2	1,733.2	2,843.7
	Soft and soda	642.9	1,097.0	1,857.2	3,097.7
	Tea	698.5	1,198.0	2,015.9	3,277.5
	Water	527.2	883.6	1,478.4	2,511.0
North	Juice	679.0	1,174.4	1,966.3	3,002.8
	Milk	618.5	1,107.4	1,841.1	2,775.0
	Soft and soda	659.3	1,154.8	1,880.2	2,952.6
	Tea	706.7	1,234.2	2,076.0	3,152.6
	Water	522.3	944.7	1,547.1	2,400.0
South	Juice	509.9	850.3	1,467.2	2,188.1
	Milk	460.2	791.3	1,356.1	2,040.2
	Soft and soda	491.9	829.1	1,455.5	2,162.8
	Tea	531.8	876.9	1,562.7	2,355.5
	Water	396.8	690.6	1,169.5	1,742.6
West	Juice	401.7	702.8	1,189.1	1,873.8
	Milk	370.1	634.5	1,082.9	1,736.5
	Soft and soda	395.3	688.9	1,161.6	1,874.0
	Tea	424.7	737.1	1,265.9	2,012.5
	Water	311.5	546.1	937.0	1,517.0
Grand total		**12,843.8**	**22,285.5**	**37,425.8**	**58,719.6**

Table G3. Consistency check. Whatever cut you make, the total revenue by year must tie to the original amount

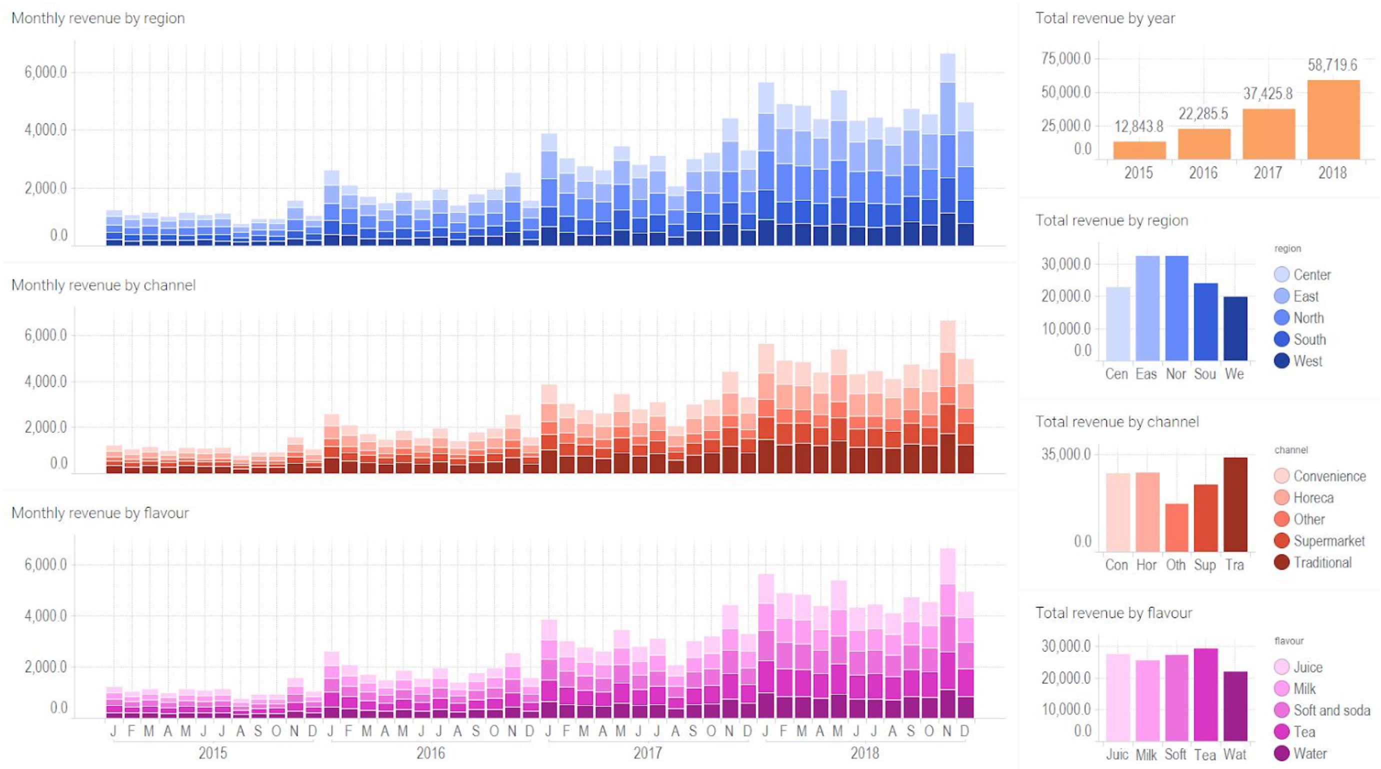


Figure G4. Data visualization

3.6. Summary

All the categorical breakdown work in the same way. We use the total multiply with the splitting weight and divide by the total weight. And that will be monotonic categorical breakdown.

If we want to perform stochastic categorical breakdown, we need to introduce a small amount of fluctuation in each splitting weight for each item. And we need to compute the total weight over the upper group of the items in order to get back the original value before breakdown.

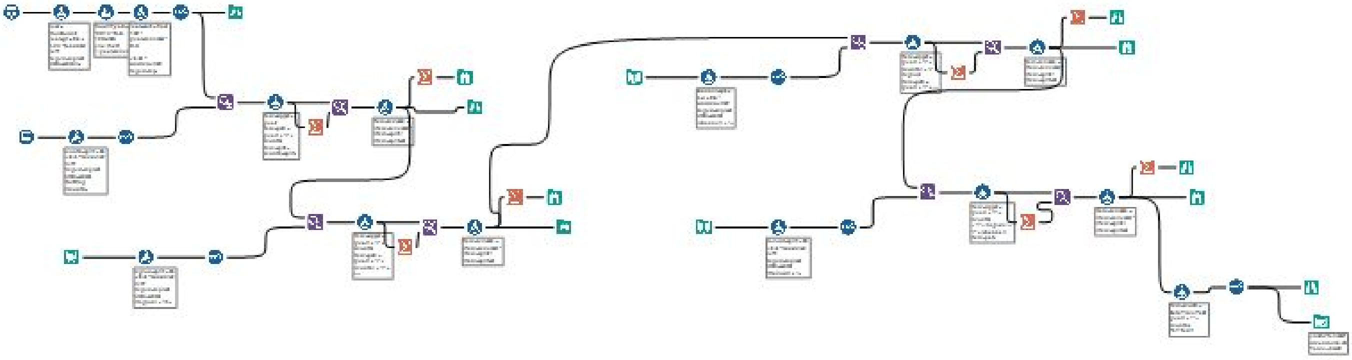


Figure G5. The alteryx workflow