R Notebook

Methodology Discussion

Input Data

Data Preparation

Modeling Approach

Final Recommendation to STC

Modeling Results

Performance Metrics

Confusion Matrix

AUC & ROC

Prediction across Choice Sets

Deciding the burn in period

After visually evaluating the runcharts for a few respondents, I have decided to use beta estimates from runs 9000 to 10,000 for all future calculations, keeping 1 to 9000 as the burn in period for stablization. Refer to burnin figure for a run-chart showing this stablization.

Q3 - Interpret your model results in regard to the attributes' effects on stated preferences.

Raw Choice Counts: Are any alternatives highly preferred by all the respondents?

A quick look into the responses of all 424 respondents pooled together reveals some high level insights. These are simple univariate looks into the data, nevertheless, they offer some insights. Histograms for each of the 5 attributes, where the height of each bar represents the number of respondents who have selected a choice-alternative containing that respective attribute level shows us that:

- Price = 0, i.e. the lowest price of $199 is the most preferred, while Price = 2 ($399) is the least preferred
- Processor = 0, i.e. 1.5Ghz is very strongly the least preferred, with the other two equally preferred
- RAM = 0, i.e. 6GB is less prefered
- For Screen, all three seem to be equally preferred
- For Brand, Brand = 0, i.e. STC seems to have a slight preference with Brand = 2, i.e. Pear following the lead

Interestingly, `choice.ID` which represents the selected alternative amongst each choice is strongly leaning towards the right, with 43% of the respondents selecting choice #3.

For all respondents, do some attributes indicate strong preferences?

We pool together the average value of the estimated beta coefficients for all the respondents for the simulation runs 9000 to 10000. Thus, we have a matrix of 424 x 14, where each row represents the average beta coefficients for each respondent for the 14 attributes (including 3 interaction terms). The best way to visualize this distribution is using the density plot below, or a boxplots as shown in the appendix. Combining these two visuals representations of the data, including the numerical summary of the coefficients in the appendix, we can gain some valuable insights into the minds of the respondents.

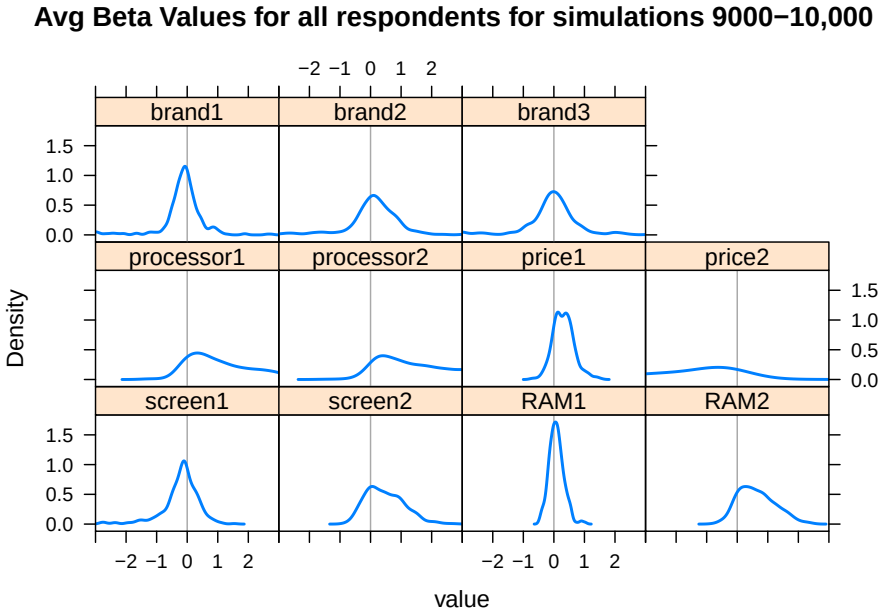


Table 1: Averaged Beta Coef for all respondents

Attribute	logOddRatio
screen1	-0.20
screen2	0.47
RAM1	0.08
RAM2	0.64
processor1	1.03
processor2	1.32
price1	0.31
price2	-2.92
brand1	-0.22
brand2	0.09
brand3	-0.30

Q4 - Study of price sensitivity by brands

Q5 - Impact of ownership of an STC product on consumer preferences

Q6 - Prediction on additional scenarios

Q7 - Attribute level partworths calculations

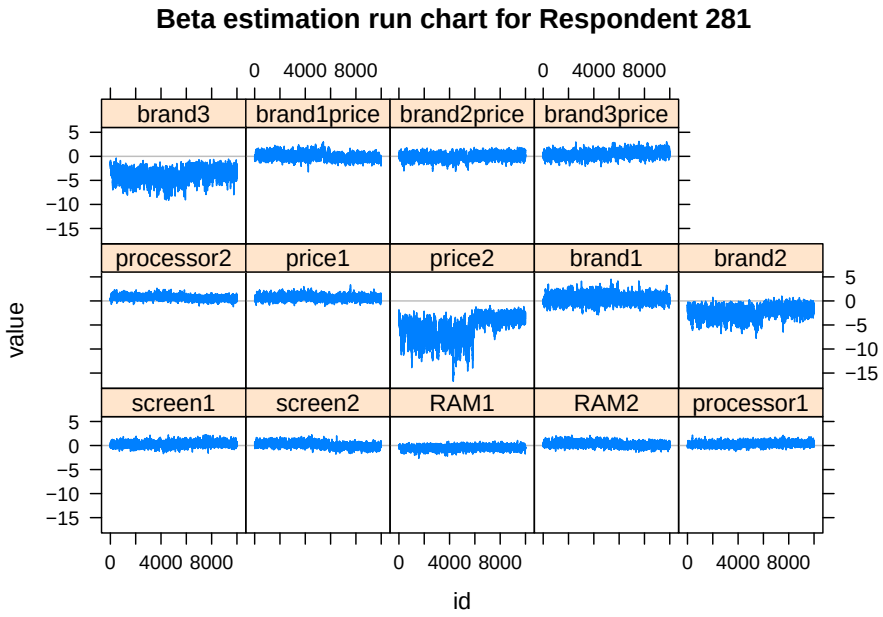
Q8 - Attribute importances for respondents

Q10 - Comparison of Performance between two models

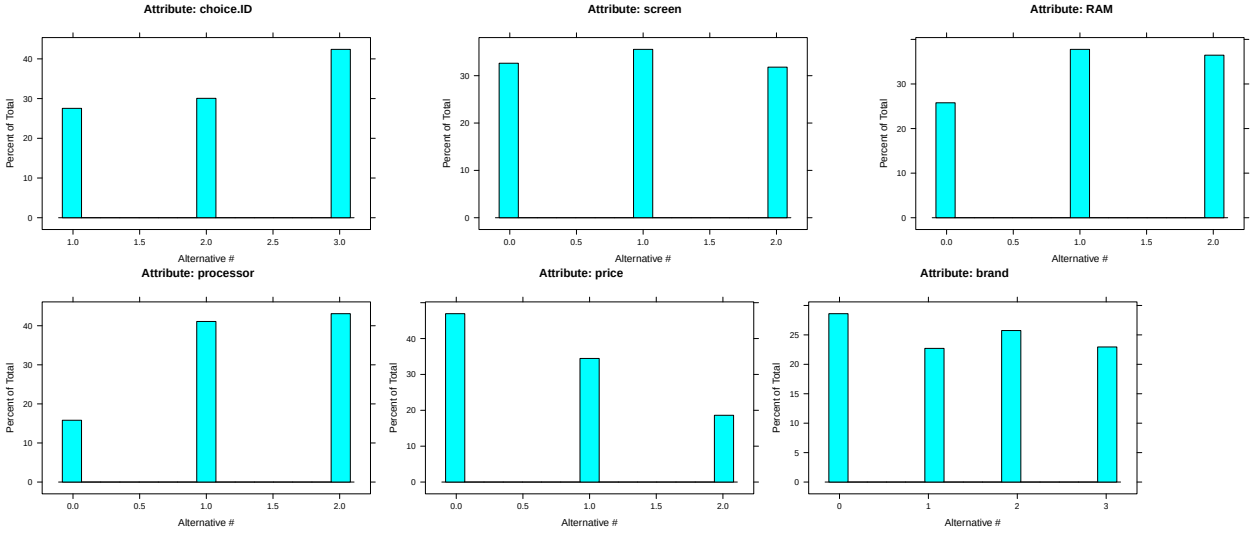
- Conf Matrix / Accuracy / AUC based discussion

Appendix

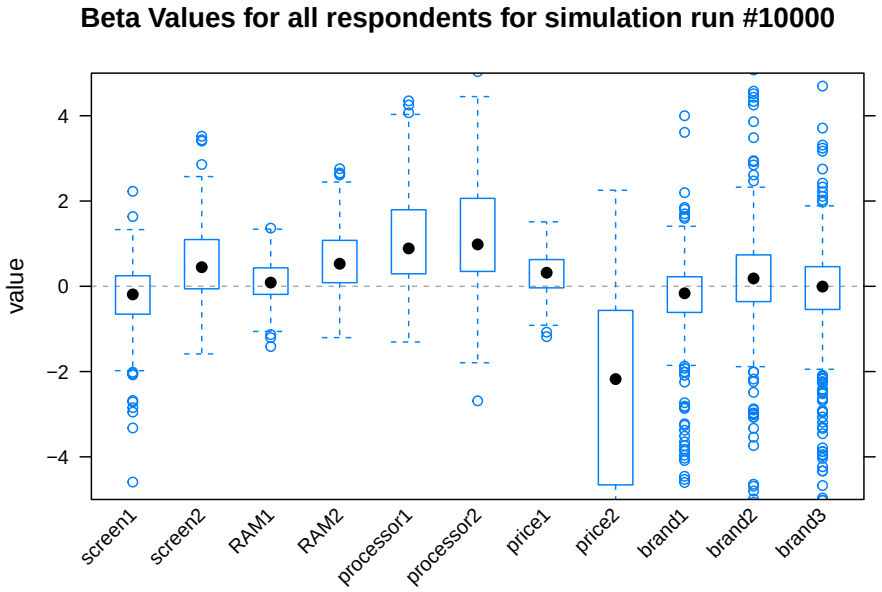
Burn-in Figure



Histograms of attribute responses for all respondents



Overall beta values for all respondents

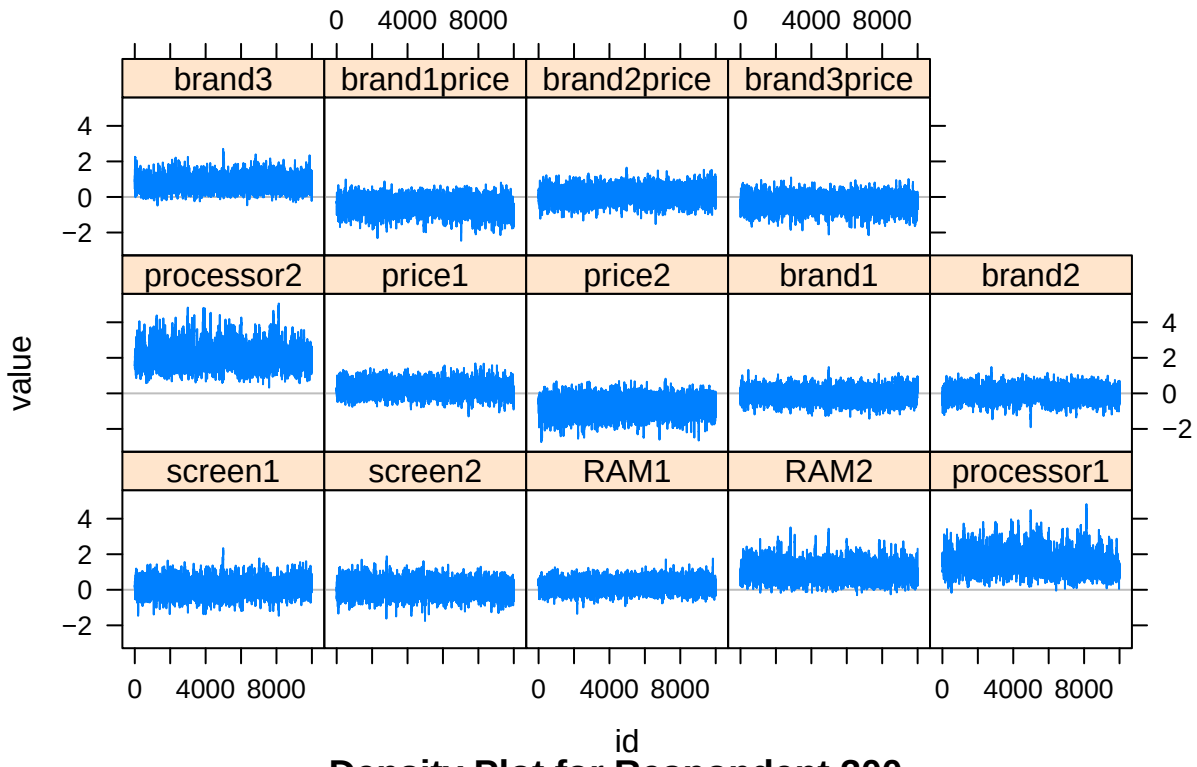


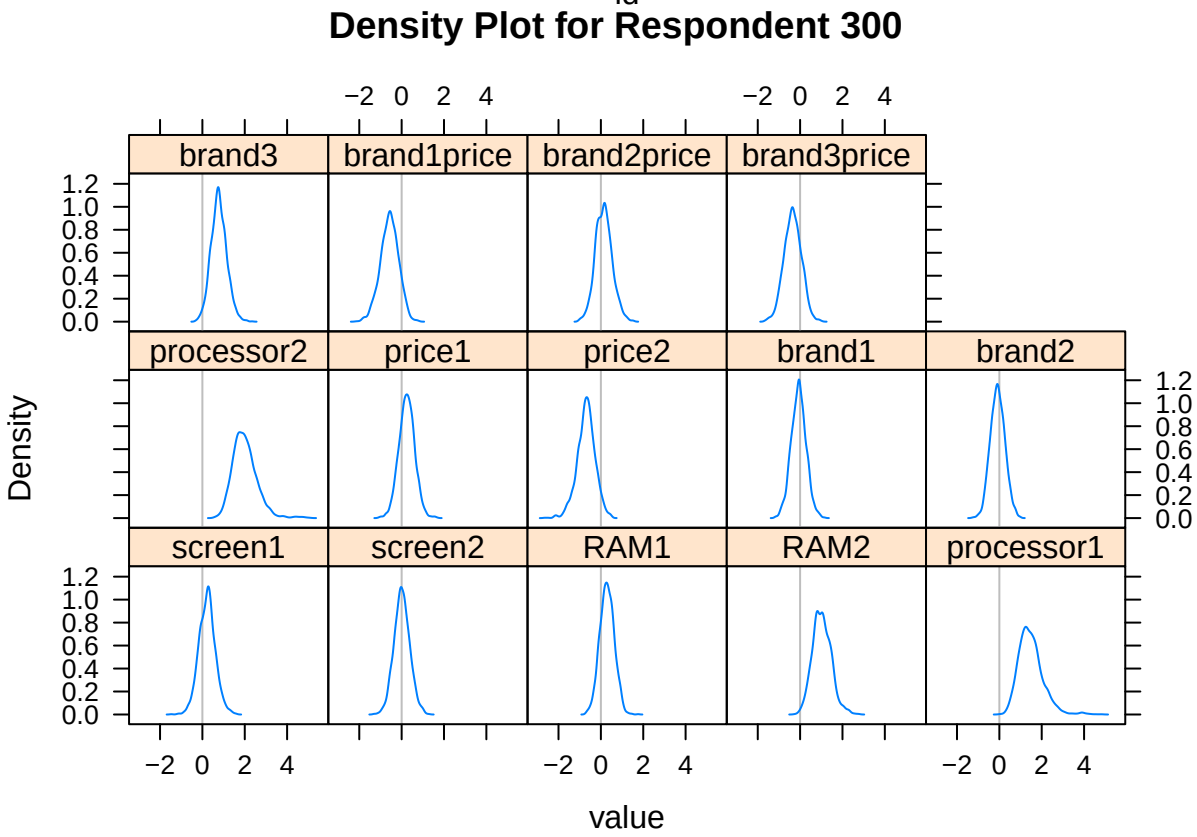
Overall beta values for all respondents

Table 2: Overall beta values for all respondents

variable	n	mean	sd	p0	p25	p50	p75	p100
brand1	424	-0.22	0.86	-4.28	-0.33	-0.097	0.13	2.77
brand2	424	0.088	1.6	-5.85	-0.26	0.13	0.58	6.65
brand3	424	-0.3	1.4	-5.24	-0.46	-0.041	0.3	5.96
price1	424	0.31	0.35	-0.74	0.084	0.31	0.52	1.55
price2	424	-2.92	2.83	-9.76	-5.1	-2.09	-0.61	2.26
processor1	424	1.03	1	-1.33	0.26	0.8	1.74	3.54
processor2	424	1.32	1.08	-1.5	0.38	1.07	2.12	3.8
RAM1	424	0.085	0.24	-0.46	-0.078	0.073	0.23	1.04
RAM2	424	0.64	0.6	-0.77	0.17	0.57	1.03	2.42
screen1	424	-0.2	0.59	-2.83	-0.42	-0.12	0.14	1.53
screen2	424	0.47	0.62	-0.84	-0.00032	0.41	0.9	2.74

```
plot_respondent_runcharts(testrun1, respondent_id = 300)
```


Beta estimation run chart for Respondent 300
brand3
brand1price
brand2price
brand3price
processor2
price1
price2
brand1
brand2
screen1
screen2
RAM1
RAM2
processor1
value
id


Density Plot for Respondent 300
brand3
brand1price
brand2price
brand3price
processor2
price1
price2
brand1
brand2
screen1
screen2
RAM1
RAM2
processor1
Density
value

Code

```
x
```