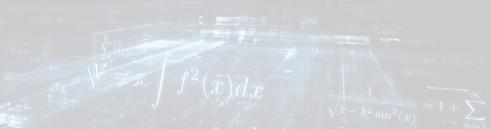


Stephane Hess stephane.hess@gmail.com

$$\lambda^{x}e^{-\lambda^{\infty}}P(x)=1$$
Making predictions: Apollo implementation



Example files

- □ We reuse the two models we worked on yesterday :
 - MNL_modeChoice_RP_base_model.r
 - MNL_modeChoice_SP_base_model.r



Generating predictions in Apollo

- apollo_prediction returns probabilities for all alternatives in the model (plus chosen) with given values for parameters
- □ We do this at the estimates but could also look at changes in parameters

```
> predictions base = apollo prediction (model, apollo probabilities, apollo inputs)
Running predictions from model using parameter estimates...
Prediction at model estimates
                      hue
Aggregate 332.00 126.00 215.00 327.00
             0 33
                    0 13
                            0 22
                                   0 33
Average
> ### keep only columns for four alternatives (removing ID, choice task and chosen)
> predictions base=predictions base[,3:6]
> summary(predictions_base)
      car
                         hus
                                           air
                                                               rail
 Min
         \cdot0 0000
                   Min.
                           \cdot0 0000
                                      Min.
                                             \cdot0 00000
                                                         Min
                                                                 \cdot 0 0000
 1st Qu.:0.2377
                   1st Qu.:0.0798
                                      1st Qu.:0.07715
                                                         1st Qu.:0.2202
 Median : 0.3536
                   Median : 0.1108
                                      Median : 0.18674
                                                         Median : 0.3096
 Mean
        :0.3320
                   Mean
                           :0.1260
                                      Mean
                                             :0.21500
                                                         Mean
                                                                 :0.3270
 3rd Qu.:0.4748
                   3rd Qu.: 0.1523
                                      3rd Qu.:0.33406
                                                         3rd Qu.: 0.4578
                           :0.6056
                                             :0.87700
 Max.
        :0.8804
                   Max.
                                      Max .
                                                         Max.
                                                                 :0.8709
```



After a 1% increase in cost for rail

```
> database$cost rail=1.01*database$cost rail
 ### Revalidate data
> apollo inputs=apollo validateInputs()
> ### Rerun predictions with the new data, and save into a separate matrix
> predictions new = apollo prediction (model, apollo probabilities, apollo inputs)
Running predictions from model using parameter estimates...
Prediction at model estimates
                     hus
             car
Aggregate 333.52 126.69 215.99 323.80
            0 33
                           0 22
Average
                    0 13
> summary(predictions new)
                        hus
      car
                                                              rail
 Min.
        :0.0000
                  Min.
                          :0.00000
                                     Min.
                                             :0.00000
                                                        Min.
                                                                :0.0000
 1st Qu.: 0.2387
                1st Qu.:0.08023
                                     1st Qu : 0.07765
                                                        1st Qu.: 0.2161
 Median : 0 3557
                 Median : 0.11135
                                     Median : 0.18806
                                                        Median : 0.3059
        :0.3335
                          :0.12669
                                             :0.21599
                                                        Mean
                                                                :0.3238
 Mean
                 Mean
 3rd Qu.: 0.4771
                  3rd Qu.: 0.15275
                                     3rd Qu.:0.33560
                                                        3rd Qu.: 0.4540
 May
        .0 8804
                  May
                          :0.60559
                                     May
                                             :0.87701
                                                        May
                                                                .0 8696
```



Demand before and after

Can also see that MNL with full set of ASC recovers market shares



Calculating elasticities

- We can approximate the elasticity by making a prediction with a very small increase in cost



SP elasticities higher than RP

```
> print(round(predictions overview .4))
Data
                          0 2780 0 0511 0 2174 0 4534
Base predictions
                          0.2780 0.0511 0.2174 0.4534
1% increase in rail costs 0.2808 0.0517 0.2193 0.4482
> ### Computing elasticities
> ### Own elasticity for rail:
> log(sum(predictions new[,4])/sum(predictions base[,4]))/log(1.01)
[1] -1.169577
> ### Cross-elasticities for other modes
> log(sum(predictions new[,1])/sum(predictions_base[,1]))/log(1.01)
[1] 0.9929864
> log(sum(predictions new[.2])/sum(predictions base[.2]))/log(1.01)
[1] 1.1788
> log(sum(predictions new[,3])/sum(predictions base[,3]))/log(1.01)
[1] 0.8663578
```







www.ApolloChoiceModelling.com

The most flexible choice modelling software (up to a probability)