Lecture 2.4. Introduction to mlogit()

- **Task 1**: Classify $S_i$’s (or objects) that vary in $\{PC_1, \ldots, PC_4\}$, as ‘A’ or ‘B’ ($Y_i = 0$ or $1$). In addition, on each classification the options, ‘A’ and ‘B’, may be assigned attributes, e.g., ‘value’ or ‘scarcity’.

- How does $Y$ depend on the person/object-specific attributes, $PC_j$, and on the option-specific attributes, e.g., ‘value’?

- **Task 2**: $S_i$’s choose between an object of category ‘A’ ($Y_i = 0$) and one of category ‘B’ ($Y_i = 1$). How does $Y$ depend on the person/chooser-specific and option/category-specific attributes?

- These tasks are formally the same.
**Task 3:** S_i’s (or objects) that vary in \{PC_1, \ldots, PC_4\} have been assigned their true labels (i.e., been classified) as ‘A’ or ‘B’ (Y_i = 0 or 1). What linear combination of \{PC_1, \ldots, PC_4\} discriminates among the labels? Is this linear discriminant function successful at classifying new S’s or objects?
Examples with more than 2 categories: With multinomial logistic regression, `mlogit()`

- Each S gives 32 self-ratings; raters then classify each S into one of 3 categories, e.g., for job training (management/tech/other)
- Each trial is characterised by the activity levels of 32 (or 3200) voxels; we wish to determine which object category (e.g., ‘face’, ‘house’) was presented on each trial.
- Use FA to reduce dimensionality of IV’s from 32 to $k$ (= 4 or 5?). Use `mlogit()` to regress a categorical DV on $k$ predictors.
Logistic Model

- Possible responses, $j$, are A, B, C ($j = 1, 2, 3$). Assign one category, e.g., ‘1’, as baseline.
- $p_{ij} = \text{Prob}(i'\text{th } S \text{ is classified as a } j)$. How to model $p_{ij}$?
- For $S_i$, the $j'$th category has a latent strength, $U_{ij}$; set $U_{i1} = 0$ (for baseline).

$$p_{i1} = \frac{1}{k} \left(1 + \sum_{l=2}^{k} e^{U_{il}}\right); \quad p_{ij} = \frac{e^{U_{ij}}}{k} \left(1 + \sum_{l=2}^{k} e^{U_{il}}\right), \quad j > 1.$$
Stimulus = \((X_1, \ldots, X_m)\)

\[ T = \sum_{j=1}^{k} e^{U_j} = 1 + \sum_{j=2}^{k} e^{U_j} \]

Categ 1 is Baseline: \(U_1 = 0\)

- **Categ 1:** \(p_1 = 1/T\)
- **Response Probs:**
  - \(p_2 = \exp(U_2)/T\)
  - \(p_k = \exp(U_k)/T\)

**Stimulus**

**Latent strength of Categ \(j\), given stim**

**Response Probs**
The multinomial logistic model – the log odds of choosing category $j$ over category 1 is linear in $U_{ij}$. To flesh it out, we assume that $U_{ij}$ is influenced by S’s traits:

\[
\begin{align*}
U_{i2} &= b_{01} + b_{11}PC_{1i} + b_{21}PC_{2i} + b_{31}PC_{3i} + b_{41}PC_{4i} \\
U_{i3} &= b_{02} + b_{12}PC_{1i} + b_{22}PC_{2i} + b_{32}PC_{3i} + b_{42}PC_{4i}
\end{align*}
\]

Thus, $p_{ij} / p_{i1} = e^{U_{ij}}$, and $\log_e (p_{ij} / p_{i1}) = U_{ij}, j > 1$. Our goal is to estimate the parameters, $\{b_{ij}\}$. 

HW-4, #1: intro to \texttt{mlogit()}

- For simplicity, consider \textbf{binary} classifications of a Person = \{PC_1, ..., PC_4\} as, e.g., ‘0’ or ‘1’. So \texttt{mlogit()} results could be obtained in the usual way with \texttt{glm(..., family=binomial,...)}. But \texttt{mlogit()} applies to multinominal and mixed models!

\[ p_i \equiv p_{i2} = \frac{e^{U_{i2}}}{1 + e^{U_{i2}}}; \text{ and } p_{i1} = 1 - p_i. \]

\[ \log(p_{i2} / p_{i1}) = \log(p_i / (1 - p_i)) = U_{i2} = b_0 + b_1 PC_{1i} + ... + b_4 PC_{4i}. \]
Sources

CITY AND REGIONAL PLANNING 775

Discrete-Choice Logit Models with R
Philip A. Viton
April 15, 2012

1. Most general, including mixed mlogit models: http://facweb.knowlton.ohio-state.edu/pviton/courses2/crp775/775-mlogit.pdf

2. Models with only participant-specific (i.e., individual-specific) variables: http://www.ats.ucla.edu/stat/r/dae/mlogit.htm

3. After installing “mlogit” and loading it with library(mlogit), use help(), as in “R > ?mFormula”. 
d = read.csv("person.option2.vars.csv")
# separate self-ratings in d0 for FA
d0 = d[, c(1:32)]; d1 = d[, c(33:37)]

# Do FA, add factor scores to d1
res1 = factanal(d0, factors=4, rotation="varimax", na.action=na.omit, scores= 'r')
#print(res1$loadings, cutoff=.4)  #to interpret PC's

# PC1 = extravert, PC2 = conscientious, 
# PC3 = agreeable, PC4 = neurotic

Data file sometimes needs to be reshaped from a ‘wide’ format (d1) to a ‘long’ format (d10). This can be done with mlogit.data().
```r
print(head(d1))

1 A TRUE -1.08 0.24 -1.23 -0.69 2.38 0.99
2 B FALSE -1.08 0.30 0.46 -0.11 0.80 -0.11
3 A FALSE 0.16 0.11 -0.64 -0.79 1.71 -1.44
4 B TRUE -0.86 -0.13 0.11 0.99 -0.42
5 A FALSE -0.13 -0.55 -0.20 -1.91 0.55
6 B TRUE -0.54 -0.05 -1.16 -0.84 0.74 1.23

d10 = mlogit.data(d1, choice = "rating2", varying = c(2:5), shape = "wide") #reshape data for mlogit()

print(head(d10))

rating2 PC1 PC2 PC3 PC4 alt attr1 attr2 chid
1.A TRUE -1.23 -0.69 2.4 0.99 A 0.9499 -1.08 1
1.B FALSE -1.23 -0.69 2.4 0.99 B -0.0061 0.24 1
2.A FALSE 0.46 -0.11 0.8 -0.11 A -0.2923 -0.95 2
2.B TRUE 0.46 -0.11 0.8 -0.11 B 0.0157 0.30 2
3.A TRUE -0.64 -0.79 1.7 -1.44 A 1.3303 0.16 3
3.B FALSE -0.64 -0.79 1.7 -1.44 B -0.0758 0.11 3
```
**Terminology**: In the syntax, ‘choice’ = ‘response’ or ‘rating’. ‘varying’ = ‘option-specific attributes or features’ ‘shape’ = ‘wide’, if each row is a trial; ‘shape’ = ‘long’, if each row is an option (within a trial)

The new cols, ‘chid’, ‘alt’ specify ‘trial’ and ‘option’.

```r
res1 = mlogit(rating2 ~ 1 | PC1 + PC2 + PC3 + PC4, d10)
print(summary(res1))

#Redo with familiar binary logistic regression using
glm(..., family = binomial, ...), data = d1
#Results are the same

res1a = glm(rating2 ~ PC1 + PC2 + PC3 + PC4, family = binomial, d1)
print(summary(res1a))
```
Note on syntax

• The mlogit() formula has 3 parts on the right-hand side:

  \texttt{mlogit(formula = rating2 \sim Part1 \mid Part2 \mid Part3, d10)}

• Part1 contains the \textbf{option-specific} attributes that might predict the DV, but do \textbf{not interact} with ‘option’; if there are no such attributes, a ‘1’ (for ‘intercept’) is put in Part1.

• If the RHS consists of a single expression, e.g., \texttt{X + Y}, this is assumed to refer to option-specific attributes. This is why, if \texttt{X} and \texttt{Y} are NOT option-specific attributes, one would have to write the RHS as \texttt{1 \mid X + Y}. 
Note on syntax

• Part2 contains the participant-specific variables that are constant across options. These variables might include personality variables (e.g., PC1), demographic variables, interactions between such person variables, etc.

• Part3 contains the option-specific attributes that do interact with ‘option’. This syntax has to be kept in mind as we construct alternative models.
In Decision Theory, the models are usually stated as models of choice, rather than classification.

Each S must choose one object from a set of objects. E.g. the set might be (i) \{local, out-of-state, overseas\} vacation; (ii) \{immediate vs delayed\} reward; (iii) \{air, train, car, other\} mode. To simplify, suppose the choice set is \{A vs B\} object.

Each object has attributes that vary within and between choice sets; e.g., price, cost, quality, risk, reward, etc. To simplify, suppose each object has 2 attributes, \((attr1, attr2)\) or \((X, Y)\).
A latent variable model of choice

• $S_i$'s $\{PC_1, \ldots, PC_4\}$ are used to classify $S_i$ as ‘A’ (‘1’) or ‘B’ (‘2’) ($Y_i = 0$ or 1). The logistic model for $p_i = \text{Prob}(Y_i = 1)$ is based on a latent variable, $U$:

$$p_i \equiv p_{i2} = \frac{e^{U_{i2}}}{1 + e^{U_{i2}}}; \text{ and } p_{i1} = 1 - p_i = \frac{1}{1 + e^{U_{i2}}}.$$ 

$$\log(p_{i2} / p_{i1}) = \log(p_i / (1 - p_i)) = \log(e^{U_{12}}) = U_{i2} = b_0 + b_1PC_{1i} + \ldots + b_4PC_{4i}.$$
What is our substantive **model** for the choice between \( A = (X_a, Y_a) \) and \( B = (X_b, Y_b) \)?

**Ans.** (i) We assign a value of the *latent* variable, \( U \) or ‘utility’, to each object in the choice set, and stipulate that \( U \) is linearly related to the *observable* attributes, \( X \) and \( Y \).

(ii) We also stipulate the function that **links** the various *latent* \( U \)’s to the *observable* choice.

In this way, observable attributes are **related to** observable choices in the model. Are the predicted relationships ‘close to’ the observed relationships? Use logistic regression.
Stochastic & Algebraic models of choice

• $U_j$ is the ‘utility’ or ‘strength’ of the $j$’th object in the choice set, $j = 1, 2, \ldots, m$.

• **Stochastic** model: $U_j$ is a random variable. The “rational” choice assn. is that the choice, $Y$, is $k$ iff $U_k$ is the maximum of the $\{U_j\}$. So

• $\text{Prob}(Y = k) = P(U_k > U_j, j \neq k)$. The details depend on the choice of probability distrns for $U_j, U_j - U_k, \ etc.$
Stochastic & Algebraic models of choice

**Algebraic model:** Let $v_j = \exp(U_j) > 0$. (Think of exponentiation as merely a device for transforming $U$ into a **positive** ‘weight’ that can enter into ratios and can be used to define probabilities.)

Set $v_1 = 1$ (i.e., $U_1 = 0$), and assume that

$$p_k \equiv \text{Prob}(Y = k) = \frac{v_k}{\sum_{j=1}^{m} v_j} = \frac{e^{U_k}}{1 + \sum_{j=2}^{m} e^{U_j}}.$$
Reconciliation

• For appropriate choices of probability distrn (notably, logistic and extreme value distrns), the stochastic model is equivalent to the algebraic model.

• The last equation is the one we use in logistic regression.
‘Interaction’ in the Choice Model

• We now have to keep track of the categorical variable, *option*, the *participant*-specific (individual-difference) variables, and *alternative/option*-specific variables (or attributes).

• Consider the *option*-specific variables, *X* and *Y*, and suppose there are 2 options, *A* and *B*. Modeling the effects of *X* and *Y* is now more complex.

• Is the effect of *X* on *U* the same for *A* and *B* objects; i.e., does it depend on ‘*option*’; i.e., is there an *X*‘*option*’ interaction?

• Does the effect of *Y* on *U* depend on ‘*option*’; i.e., is there an *Y*‘*option*’ interaction?
Interactions (cont’d)

• We have just defined $Option*Attribute$ interactions for each attribute. There also might be $Option*PC$ interactions, and $PC*Attribute$ interactions.

• First, a brief review of interaction.

• Then consider homogeneous, heterogeneous and hybrid models of choice, paying attention to the 3-part syntax of mlogit().
Brief review of ‘interaction’

• Variables are quantitative or dummy-coded (to be 0/1 variables and, therefore, act like quantitative variables)

• **Algebraic** symptom: $b_3 \neq 0$, where $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_1X_2$

• **Graphical** symptom: Are the lines parallel? Suppose $X_2$ is a 0/1 variable.

• When $X_2 = 0$, $Y = b_0 + b_1X_1$; slope = $b_1$.

• When $X_2 = 1$, $Y = (b_0 + b_2) + (b_1 + b_3)X_1$; slope = $b_1 + b_3$. 
Generic description of ‘interaction’

• $A (= 1, 2)$ and $B$ (e.g., quant) are predictors; $Y$ is DV. Is the $A*B$ interaction sig? “No”, if the effect of $A$ on $Y$ the same at each level of $B$. Or: “No”, if the effect of $B$ on $Y$ the same at each level of $A$.

** Describe effect of $B$ on $Y$ when $A = 1$

** Describe effect of $B$ on $Y$ when $A = 2$

** Compare/contrast the above 2 descriptions

* Add any other ‘interesting’ features; e.g., any non-linear effects, or is the interaction due to only 1 group, or crossover effect, ...?
Ex: health vs stress: decreasing function if S is ‘not innoculated’, but flat if S is ‘innocul’. Health = $b_0 + b_1 \cdot Str + b_2 \cdot Innoc + b_3 \cdot Str \cdot Innoc$. When $Innoc = 0$, Health = $b_0 + b_1 \cdot Str$. When $Innoc = 1$, Health = $(b_0 + b_2) + (b_1 + b_3) \cdot Str$. So, perhaps $b_1 < 0$, and $(b_1 + b_3) \approx 0$,
Group*lin(age) n.s.;
Group*quad(age) sig.
Homogeneous models of choice

- The effects of attributes, $X$ and $Y$, on $U$ are the **same** for both options, A and B. Let $s_i$ be any individual-differences variable (e.g., age, extraversion) for $S_i$. The utilities for A and B are $U_{ia}$ and $U_{ib}$.

- $U_{ia} = s_{ia} + c_1 X_a + c_2 Y_a$
- $U_{ib} = s_{ib} + c_1 X_b + c_2 Y_b$
- Prob(Choice is B) is a function of the **difference**, $U_{ib} - U_{ia} = c_1 (X_b - X_a) + c_2 (Y_b - Y_a) + s_i'$
mlogit(formula = rating2 ~ attr1 + attr2, data = d10, method = "nr", print.level = 0)

Coefficients :

|              | Estimate | Std. Error | t-value | Pr(>|t|)  |
|--------------|----------|------------|---------|-----------|
| B:(intercept) | 0.74314  | 0.22744    | 3.2675  | 0.001085 ** |
| attr1        | 1.77804  | 0.26637    | 6.6752  | 2.469e-11 *** |
| attr2        | -0.20329 | 0.21379    | -0.9509 | 0.341674 |

\[ U_{ia} = s_{ia} + c_1X_a + c_2Y_a + e \]
\[ U_{ib} = s_{ib} + c_1X_b + c_2Y_b + e. \]

In the above, it is assumed that \( s_{ia} = s_a \) and \( s_{ib} = s_b \) do not vary across Ss. The estimates are: \( c_1 = 1.78 \ (p < .001) \), \( c_2 = -0.20 \ (n.s.) \); \( s_b - s_a = 0.74 \ (p < .01) \). Effect of Y (attr2) is not sig.
• In sum, for the homogeneous model, compute the difference between options, A and B, in attr1 and in attr2. Use these differences to predict choice from \( D = c_1 (X_b - X_a) + c_2 (Y_b - Y_a) + s' \).

• In the **heterogeneous** model, the effects of attributes, \( X \) and \( Y \), on \( U \) are **different** for the 2 options.
  - \( U_{ia} = s_{ia} + c_{1a}X_a + c_{2a}Y_a \)
  - \( U_{ib} = s_{ib} + c_{1b}X_b + c_{2b}Y_b \), and
  - \( D' = (c_{1b}X_b - c_{1a}X_a) + (c_{2b}Y_b - c_{2a}Y_a) + s' \).
Hybrid model

Suppose effect of $X$, but not $Y$, is same for A and B.
Then $c_{1a} = c_{1b} = c_1$.

$D' = c_1(X_b - X_a) + (c_{2b}Y_b - c_{2a}Y_a) + s'$

$= c_1(X_b - X_a) + c_{2b}(Y_b - Y_a) + (c_{2b} - c_{2a})Y_a + s'$.

To represent this model in a \textbf{GLM}, we would need the 2 attribute-differences plus a 3\textsuperscript{rd} term with one (either one) of the raw attribute levels, $Y_a$ or $Y_b$.

\begin{verbatim}
d1$diff1 = d1$attr1.A - d1$attr1.B
d1$diff2 = d1$attr2.A - d1$attr2.B

res4a = glm(rating2 ~ diff1 + diff2 + attr2.B + PC1 + PC2 + PC3 + PC4, family = binomial, d1)
\end{verbatim}
mlogit(formula = rating2 ~ attr1 | PC1 + PC2 + PC3 + PC4 | attr2, data = d10, method = "nr", print.level= 0)

Coefficients:

|                | Estimate | Std. Error | t-value | Pr(>|t|)   |
|----------------|----------|------------|---------|------------|
| B:(intercept)  | -0.212302| 0.315613   | -0.6727 | 0.5011592 |
| attr1          | 1.424112 | 0.308923   | 4.6099  | 4.028e-06 *** |
| B:PC1          | 0.712494 | 0.227206   | 3.1359  | 0.0017133 ** |
| B:PC2          | 0.912937 | 0.252104   | 3.6213  | 0.0002932 *** |
| B:PC3          | 0.411245 | 0.242581   | 1.6953  | 0.0900207 . |
| B:PC4          | 0.027927 | 0.223295   | 0.1251  | 0.9004700 |
| A:attr2        | -1.114559| 0.320151   | -3.4814 | 0.0004989 *** |
| B:attr2        | 4.272483 | 0.803135   | 5.3198  | 1.039e-07 *** |

\[ U_{ia} = s_{ia} + c_{1a}X_{a} + c_{2a}Y_{a} \]
\[ U_{ib} = s_{ib} + c_{1b}X_{b} + c_{2b}Y_{b}. \]

In the above, \( c_{1a} = c_{1b} = 1.42; c_{2a} = -1.11, c_{2b} = 4.27 \) (all highly sig). So ‘effect of Y’ now sig!
Lecture 2.4. Interactions in \texttt{mlogit()}; Cluster Analysis

- \textit{Option-specific Attribute} * \textit{Option} interaction. To specify these, use Parts 1 and 3 of the right-hand side of the formula in the model syntax.

- What about \textit{PC} * \textit{Option} and \textit{PC} * \textit{Attribute} interactions?

- \textit{PC} * \textit{Option} is already accounted for as the \textit{main} effect of \textit{PC} on Prob(Choosing \textit{B} over \textit{A}).

- \textit{PC} * \textit{Attribute} interactions need to be defined in the data file, and then included as \textit{person-specific variables} in Part 2 of the formula.

- \textbf{Random} intercept models
- Assumption of \textbf{Independence of Irrelevant Alternatives}
Model syntax

• We’ve seen above that the `mlogit()` formula has 3 parts on the right-hand side:

```r
mlogit(formula = rating ~ Part1 | Part2 | Part3, d10)
```

• Part1 contains the option-specific attributes that might predict the DV but do not interact with ‘option’; if there are no such attributes, a ‘1’ (for ‘intercept’) is put in Part1. Part2 contains the participant-specific variables that are constant across options. Part3 contains the option-specific attributes that do interact with ‘option’. This allows us to define `attribute * option` interactions.
The regression coefficients in a hybrid model in which the effect of 1 option-specific attribute, Attr1, is the same for all four options, but that of Attr2 varies across options. Option 1 is set as the reference level; i.e., $U_1$ is set at 0.

.rating4 \sim \text{attr1} \mid \text{PC1} + \text{PC2} \mid \text{attr2}$

<table>
<thead>
<tr>
<th>Utility for $j$'th option on a trial, $j = 1, ..., 4$</th>
<th>Person-specific variables</th>
<th>Option-specific variables</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$PC_1$</td>
<td>$PC_2$</td>
</tr>
<tr>
<td>$U_1$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$U_2$</td>
<td>$b_{21}$</td>
<td>$b_{22}$</td>
</tr>
<tr>
<td>$U_3$</td>
<td>$b_{31}$</td>
<td>$b_{32}$</td>
</tr>
<tr>
<td>$U_4$</td>
<td>$b_{41}$</td>
<td>$b_{42}$</td>
</tr>
</tbody>
</table>
Example: mlogit(formula = rating2 ~ attr1 | PC1 + PC2 + PC3 + PC4 | attr2, data = d10, method = "nr", print.level= 0)

Coefficients:

|          | Estimate | Std. Error | t-value | Pr(>|t|)   |
|----------|----------|------------|---------|-----------|
| B:(intercept) | -0.212302 | 0.315613 | -0.6727 | 0.5011592 |
| attr1     | 1.424112  | 0.308923  | 4.6099  | 4.028e-06 *** |
| B:PC1     | 0.712494  | 0.227206  | 3.1359  | 0.0017133 ** |
| B:PC2     | 0.912937  | 0.252104  | 3.6213  | 0.0002932 *** |
| B:PC3     | 0.411245  | 0.242581  | 1.6953  | 0.0900207 . |
| B:PC4     | 0.027927  | 0.223295  | 0.1251  | 0.9004700 |
| A:attr2   | -1.114559 | 0.320151  | -3.4814 | 0.0004989 *** |
| B:attr2   | 4.272483  | 0.803135  | 5.3198  | 1.039e-07 *** |

\[
U_{ia} = s_{ia} + c_{1a} X_a + c_{2a} Y_a \\
U_{ib} = s_{ib} + c_{1b} X_b + c_{2b} Y_b.
\]

In the above, \( c_{1a} = c_{1b} = 1.42; c_{2a} = -1.11, c_{2b} = 4.27 \) (all highly sig). So ‘effect of Y’ now sig!
Suppose we have 2 options, A = a ‘local’, and B = an ‘overseas’ vacation, and $PC = neuroticism$. Maybe utility, $U$, is independent of $PC$ for A (local option), but $U$ decreases as $PC$ increases for B (overseas option). This appears to be a $PC*option$ interaction.

However, each S has a fixed level of $PC$, and evaluates the 2 options with that $PC$ level. The above insight implies that, if $S_2$ has a greater $PC$ level than $S_1$, then $S_2$ is less likely to choose B. But this is merely the main effect of $PC$ on $Prob(Choosing\ B)$. 
This may be why `mlogit()` output labels the Coefficients as if they were interactions: e.g., `B:(intercept)`, `B:PC1`, etc., but `glm()` doesn’t.

```r
mlogit(formula = rating2 ~ 1 | PC1 + PC2 + PC3 + PC4, data = d10, method = "nr", print.level = 0)

Coefficients:

| Estimate | Std. Error | t-value  | Pr(>|t|) |
|----------|------------|----------|----------|
| B:(intercept) | -0.6594375 | 0.1572017 | -4.1949  | 2.731e-05 *** |
| B:PC1     | 0.7154716  | 0.1694285 | 4.2229   | 2.412e-05 *** |
```

```r
glm(formula = rating2 ~ PC1 + PC2 + PC3 + PC4, family = binomial, data = d1)

Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| (Intercept) | -0.659438  | 0.157198 | -4.195   | 2.73e-05 *** |
| PC1      | 0.715472   | 0.169424 | 4.223    | 2.41e-05 *** |
```
**PC*attr?**

- Use choice model to give **meaning** to *PC*attr. Use **reshaping** of data file, and **syntax** of the formula in `mlogit()`, to define this interaction.

- Choice is determined by:
  \[ D = c_1 (X_b - X_a) + c_2 (Y_b - Y_a) + s'. \]

- Possible meaning of PC*X is that the effect of \((X_b - X_a)\) on \(D\) depends on PC, i.e., that \(c_1\) depends on PC. To capture this effect, define a product term, e.g., \(PCX = PC*\text{scale}(X_b - X_a)\), and treat \(PCX\) as an individual-specific variable, like \(PC\).

- Define \(PCX\) in original, ‘wide’ data file, \(d1\), then reshape \(d1\) to \(d11\).
IV’s are: PC1, PC2; attr1, attr2; PC1*attr

d1$diff1 = d1$attr1.A - d1$attr1.B

d1$diff2 = d1$attr2.A - d1$attr2.B

d1$PC1att1 = d1$PC1*scale(d1$diff1)

d1$PC1att2 = d1$PC1*scale(d1$diff2)

d11 = mlogit.data(d1, choice = "rating2", varying = c(2:5), shape = "wide")

res7 = mlogit(rating2 ~ attr1 + attr2 | PC1 + PC2 + PC1att1 + PC1att2, d11)

res8 = mlogit(rating2 ~ attr1 | PC1 + PC2 + PC1att1 + PC1att2 | attr2, d11)
mlogit(formula = rating2 ~ attr1 + attr2 | PC1 + PC2 + PC1att1 + PC1att2, data = d11, method = "nr", print.level = 0)

Coefficients:

|                | Estimate | Std. Error | t-value | Pr(>|t|) |
|----------------|----------|------------|---------|----------|
| B:(intercept)  | 0.51286  | 0.25021    | 2.0497  | 0.040394 * |
| attr1          | 1.62071  | 0.27771    | 5.8359  | 5.351e-09 *** |
| attr2          | -0.22524 | 0.23403    | -0.9624 | 0.335827   |
| B:PC1          | 0.52266  | 0.19109    | 2.7351  | 0.006235 ** |
| B:PC2          | 0.96650  | 0.22013    | 4.3906  | 1.130e-05 *** |
| B:PC1att1      | -0.25901 | 0.22583    | -1.1469 | 0.251406   |
| B:PC1att2      | 0.14785  | 0.18033    | 0.8199  | 0.412280   |

Describe main effects and **PC** * Attr interactions, when there is no Attr * Option interaction.
mlogit(formula = rating2 ~ attr1 | PC1 + PC2 + PC1att1 + PC1att2 | attr2, data = d11, method = "nr", print.level = 0)

Coefficients:

|                | Estimate | Std. Error | t-value | Pr(>|t|)   |
|----------------|----------|------------|---------|-----------|
| B:(intercept)  | -0.15743 | 0.31740    | -0.4960 | 0.6198891 |
| attr1          | 1.52102  | 0.31315    | 4.8572  | 1.191e-06 *** |
| B:PC1          | 0.60836  | 0.22700    | 2.6800  | 0.0073624 ** |
| B:PC2          | 0.94097  | 0.25595    | 3.6764  | 0.0002366 *** |
| B:PC1att1      | -0.46832 | 0.24468    | -1.9140 | 0.0556245 . |
| B:PC1att2      | 0.12692  | 0.21119    | 0.6010  | 0.5478610  |
| A:attr2        | -1.22041 | 0.33510    | -3.6419 | 0.0002707 *** |
| B:attr2        | 4.32668  | 0.81121    | 5.3336  | 9.629e-08 *** |

Describe main effects and **PC * Attr** interactions, when there is an **Attr * Option** interaction.
Mixed Models

• Is model fit improved by assuming that certain parameters vary randomly across Ss? An obvious candidate is “B:(intercept)”, the generic preference or bias for B over A when all IV’s = 0, labeled $s_{ib} - s_{ia}$ earlier.

• My results are mixed – one random-effects model converged but the se(estimate) was ‘large’, etc. At least we can take note of the syntax involving `rpar()`. 
Illustrations of the rpar() syntax

res6 = mlogit(rating2 ~ attr1 + attr2 | PC1 + PC2 + PC3 + PC4, d10, rpar = c("B: (intercept)" = "n"), R=500, halton=NA)

res6a = mlogit(rating2 ~ attr1 + attr2 | PC1 + PC2 + PC3 + PC4, d10, rpar = c("B: (intercept)" = "n", "B:PC2" = "n"), R=500, halton=NA)

'R' specifies the precision with which certain probs are estimated; 'halton' specifies the quasi-random numbers to be used in the simulations.
res6 = mlogit(rating2 ~ attr1 + attr2 | PC1 + PC2 + PC3 + PC4, d10, rpar = c("B:(intercept)" = "n"), R=500, halton=NA)

Coefficients:

|                  | Estimate  | Std. Error | t-value | Pr(>|t|) |
|------------------|-----------|------------|---------|----------|
| B:(intercept)    | 0.485511  | 0.283633   | 1.7118  | 0.0869410 |
| attr1            | 1.602107  | 0.476773   | 3.3603  | 0.0007785 ***|
| attr2            | -0.193927 | 0.244163   | -0.7942 | 0.4270501 |
| B:PC1            | 0.598524  | 0.258195   | 2.3181  | 0.0204432 * |
| B:PC2            | 0.949654  | 0.392366   | 2.4203  | 0.0155066 * |
| B:PC3            | 0.372120  | 0.260436   | 1.4288  | 0.1530516 |
| B:PC4            | 0.077960  | 0.187779   | 0.4152  | 0.6780174 |
| sd.B:(intercept) | 0.019693  | 40.001911  | 0.0005  | 0.9996072 |

s.e.(estimate) = 40, suspiciously large! It probably means that the 'intercept' is fixed.
res6a = mlogit(rating2 ~ attr1 + attr2 | PC1 + PC2 + PC3 + PC4, d10, rpar = c("B:(intercept)" = "n", "B:PC2" = "n"), R=500, halton=NA)

Coefficients:

|                | Estimate | Std. Error | t-value | Pr(>|t|) |
|----------------|----------|------------|---------|----------|
| B:(intercept)  | 0.4855171| 0.2993767  | 1.6218  | 0.10485  |
| attr1          | 1.6021110| 0.4991878  | 3.2094  | 0.00133  **|
| attr2          | -0.1939353| 0.2454105  | -0.7902 | 0.42938  |
| B:PC1          | 0.5985269| 0.2659107  | 2.2509  | 0.02439  *|
| B:PC2          | 0.9496553| 0.4040715  | 2.3502  | 0.01876  *|
| B:PC3          | 0.3721278| 0.2610839  | 1.4253  | 0.15407  |
| B:PC4          | 0.0779633| 0.1898552  | 0.4106  | 0.68133  |
| sd.B:(intercept)| 0.0197247| 0.1078195  | 0.0005  | 0.99961  |
| sd.B:PC2       | 0.0031637| 54.9208995 | 0.0001  | 0.99995  |

Again, s.e.(estimate) = 40, 55, suggesting that the parameters are fixed.
**IIA.** Most of the issues already considered for the binary DV generalise naturally to the multinomial case. One exception is the assumption of the *independence of irrelevant alternatives* (IIA).

Consider the equations (ignoring the S index, $i$),

\[
\log(e(p_2/p_1)) = U_2, \\
\log(e(p_3/p_1)) = U_3.
\]

In the basic logistic model considered so far, it is **assumed** that the probability of choosing $alt2$ over $alt1$ does not depend on which other alternatives are in the choice set. This is the **IIA** assumption.
IIA

• Clearly IIA is not relevant with only 2 options.

• With 3 or more options, $alt1$, $alt2$, ..., we can estimate, e.g., $\log_e(p_2/p_1)$ in 2 ways
  – Use all the data in the study
  – Use only those trials on which $alt1$ or $alt2$ is chosen

• The estimates of $\log_e(p_2/p_1)$ in the 2 analyses should be similar under IIA. IIA can be tested with the Hausman-McFadden test.

res1 = mlogit(rating3 ~ 1 | PC1 + PC2, d10)
res2 = mlogit(rating3 ~ 1 | PC1 + PC2, d10, alt.subset = c("1","2"))
res12 = hmftest(res1, res2)
print(res12)  [Use HW-4, #4 as example!]
Recommended Reading

Appendix 1: $R^2$ and $\chi^2$ in mlogit() output

http://www.ats.ucla.edu/stat/mult_pkg/faq/general/Psuedo_RSquareds.htm

The $\chi^2$ statistic is equal to $2*[\log(L(M1)) – \log(L(M0))]$, where $L(M0)$ is the likelihood of the ‘intercept only’ model with no predictors, and $L(M1)$ is the the likelihood of the ‘full’ model with all the predictors. The interpretation of $R^2$ is more complicated.

When analyzing data with a logistic regression, an equivalent statistic to R-squared does not exist. The model estimates from a logistic regression are maximum likelihood estimates arrived at through an iterative process. They are not calculated to minimize variance, so the OLS approach to goodness-of-fit does not apply. However, to evaluate the goodness-of-fit of logistic models, several pseudo R-squareds have been developed. These are "pseudo" R-squareds because they look like R-squared in the sense that they are on a similar scale, ranging from 0 to 1 (though some pseudo R-squareds never achieve 0 or 1) with higher values indicating better model fit, but they cannot be interpreted as one would interpret an OLS R-squared and different pseudo R-squareds can arrive at very different values. Note that most software packages report the natural logarithm of the likelihood due to floating point precision problems that more commonly arise with raw likelihoods. ... Note that the pseudo R-squareds vary greatly from each other within the same model.
The interpretation of an OLS R-squared is relatively straightforward: "the proportion of the total variability of the outcome that is accounted for by the model". In building a model, the aim is usually to predict variability. The outcome variable has a range of values, and you are interested in knowing what circumstances correspond to what parts of the range. If you are looking at home values, looking at a list of home prices will give you a sense of the range of home prices. You may build a model that includes variables like location and square feet to explain the range of prices. If the R-squared value from such a model is .72, then the variables in your model predicted 72% of the variability in the prices. So most of the variability has been accounted for, but if you would like to improve your model, you might consider adding variables. You could similarly build a model that predicts test scores for students in a class using hours of study and previous test grade as predictors. If your R-squared value from this model is .75, then your model predicted 75% of the variability in the scores.

Though you have predicted two different outcome variables in two different datasets using two different sets of predictors, you can compare these models using their R-squared values: the two models were able to predict similar proportions of variability in their respective outcomes, but the test scores model predicted a slightly higher proportion of the outcome variability than the home prices model. Such a comparison is not possible using pseudo R-squareds.
If two logistic models, each with \( N \) observations, predict different outcomes and both predict their respective outcomes perfectly, then the Cox & Snell pseudo R-squared for the two models is \((1-L(M_{Intercept})^2/N)\). However, this value is not the same for the two models. The models predicted their outcomes equally well, but this pseudo R-squared will be higher for one model than the other, suggesting a better fit. Thus, these pseudo R-squareds cannot be compared in this way.

**Intention** - Recall that OLS minimizes the squared differences between the predictions and the actual values of the predicted variable. This is not true for logistic regression. The way in which R-squared is calculated in OLS regression captures how well the model is doing what it aims to do. Different methods of the pseudo R-squared reflect different interpretations of the aims of the model. In evaluating a model, this is something to keep in mind.

For example, Efron's R-squared and the Count R-squared evaluate models according to very different criteria: both examine the residuals--the difference between the outcome values and predicted probabilities--but they treat the residuals very differently. Efron's sums the squared residuals and assesses the model based on this sum. Two observations with small differences in their residuals (say, 0.49 vs. 0.51) will have small differences in their squared residuals and these predictions are considered similar by Efron's. The Count R-squared, on the other hand, assesses the model based solely on what proportion of the residuals are less than .5. Thus, the two observations with residuals 0.49 and 0.51 are considered very differently: the observation with the residual of 0.49 is considered a "correct" prediction while the observation with the residual of 0.51 is considered an "incorrect" prediction. When comparing two logistic models predicting different outcomes, the intention of the models may not be captured by a single pseudo R-squared, and comparing the models with a single pseudo R-squared may be deceptive.
$R^2$ and $\chi^2$ in output

For some context, we can examine another model, Model 2, predicting the same variable in the same dataset as the model above (Model 1), but with one added variable.

All of the pseudo R-squareds reported here agree that Model 2 better fits the outcome data than Model 1. While pseudo R-squareds cannot be interpreted independently or compared across datasets, they are valid and useful in evaluating multiple models predicting the same outcome on the same dataset. In other words, a pseudo R-squared statistic without context has little meaning. A pseudo R-squared only has meaning when compared to another pseudo R-squared of the same type, on the same data, predicting the same outcome. In this situation, the higher pseudo R-squared indicates which model better predicts the outcome.