# Precept Seven: Mixture Models and the EM Algorithm

Rebecca Johnson

March 28th, 2017

# Outline

- Replication/psets check-in
- Mixture model example: Middle-Inflated Ordered Probit (MiOP) model with views on EU membership
- Shift to EM algorithm with different mixture: wines from distinct cultivars (plants) in Italy
  - Move from mixture of $k = 2$ *univariate* normals to mixture of $k = 3$ *multivariate* normals
  - Practice coding EM algorithm into R to gain intuition on the algorithm

# Inflation models as mixture models

- In precept 6, Ian took us through an example of a *zero-inflated negative binomial* (if a logit and negative binomial model mated and gave birth to a *zinb*!)
  - *Example*: count of male 'satellites' around a female horseshoe crab
  - Counts are a *mixture* of two processes, each modeled using a different distribution:
    - Whether or not the female attracts *any* satellites...modeled using a Bernoulli distribution that draws $Z_i = 0$ or $Z_i = 1$
    - Conditional upon attracting satellites, the count she attracts...modeled using a Negative Binomial distribution that draws from positive integers *including* 0

# Inflation models as mixture models

- On problem set 5, we have an example of a *zero-inflated poisson* (if a logit (or probit) mated with a Poisson and gave birth to a *zip*!)
    - *Example*: the count of speeches critical of the Iraq War that Republican house members give in a particular month
    - Counts are a *mixture* of two processes, each modeled using a different distribution:
        - Whether or not there are any speeches given...modeled using a Bernoulli distribution that draws $Z_i = 0$ or $Z_i = 1$
        - Conditional upon any speeches, the count of speeches...modeled using a Poisson distribution that draws from positive integers *including* 0

# This framing of the two zero-inflated models should look familiar from Monday's lecture...

- *Mixture models* (sometimes called finite mixture models): we assume that each observation is generated from one of $k$ clusters/distributions
- *Common notation*:
  - Latent variable for which distribution/cluster: $Z$ before estimated; $z$ after estimated; $z_i$ indicates which distribution/cluster observation $i$ comes from
  - Number of distributions/clusters to choose from: $k$
  - Putting these together: $z_i \in \{1, 2, \ldots k\}$

## This framing of the two zero-inflated models should look familiar from Monday's lecture...

- In previous examples, $k = 2$, allowing us to model $k$ using distributions for binary outcomes like logit and probit
- In other examples, $k > 2$, meaning we need to switch to a distribution that allows us to draw from more than two categories...[1]
  - **Multinomial distribution**, where $1 =$ number of trials and $\pi =$ probability of choosing category $1, 2 \dots k$:

$$z_i | \pi \sim Multinomial(1, \pi)$$

[1] More formally, we can think of the Bernoulli distribution behind the logit model as a special case of a Multinomial when the number of trials $= 1$ and $k = 2$

# Mixture models we've seen: happen to be models where we've assumed observations are drawn from one of two categories

- **Old faithful and height by sex examples:** mixture of normals (Gaussian mixture) with $k = 2$
  - *What's 'mixed'?*: same distribution (normal) but each of the $k = 2$ normal distributions has a different mean and variance
  - *Which parameters can we estimate?*: $\{\mu_1, \mu_2, \Sigma_1, \Sigma_2, \pi\}$
- **Horseshoe crab and count of speeches examples:** mixture of two distributions with $k = 2$
  - *What's 'mixed'?*: different distributions: a distribution that explains zero's and a distribution that explains counts that include zero's (negative binomial; poisson)
  - *Which parameters can we estimate?*: $\{\beta$ for logit or probit; $\gamma$ for negative binomial or poisson, $\pi\}$
- **Voting on trade bills example**: mixture of regression models with $k = 2$ (Stoper-Samuelson theory v. Ricardo-Viner theory)

## Mixture models: examples where $k > 2$

- If Garip (2012) had estimated membership in one of the four migration clusters using EM algorithm rather than *k-means* clustering
- Multivariate normal example we'll turn to later where $k = 3$ distinct plants used to grow wine
- Many others! (extracting dominant $k$ dominant colors from images, modeling ancestry, etc.)

# New mixture model for today's precept...

- **Middle-inflated ordered probit model** (MiOP)

  *Bagozzi, Benjamin E., Bumba Mukherjee, and R. Michael Alvarez. A mixture model for middle category inflation in ordered survey responses. Political Analysis (2012): 369-386.*
- Why we're covering:
  1. Reiterates general ideas behind zero-inflated model you'll derive/estimate in Pset 5 because builds on general intuition behind zero inflation
  2. Useful for applied survey work using Likert-type scales

# Motivation for MiOP: Eurobarometer poller takes a trip to Vilnius, Lithuania in 2002

EUROBAROMETER

# Motivation for MiOP: Eurobarometer poller takes a trip to Vilnius, Lithuania in 2002

- *Interviewer*: "Generally speaking, do you think that Lithuania's membership in the European Union would be a good thing, a bad thing, or neither good nor bad? (or you can choose *do not know*)"
  - *Informed Vilnius resident 1:* a good thing!
  - *Informed Vilnius resident 2:* neither good or bad! I can see the benefits of easier migration, but I also think we benefit from having *litas* and that switching to the Euro might induce inflation
  - *Uninformed Vilnius resident who is willing to admit he or she is uninformed:* don't know!
  - *Uninformed Vilnius resident who is* **not** *willing to admit he or she is uninformed:* neither good or bad! (*while thinking: I don't want to choose 'do not know' because that will show I'm clueless about the EU*)

# Focusing on the neither good nor bad category, how do we distinguish between...?

1. *Informed Vilnius resident 2:* neither good or bad! I can see the benefits of easier migration, but I also think we benefit from having *litas* and that switching to the Euro might induce inflation
2. *Uninformed Vilnius resident who is* **not** *willing to admit he or she is uninformed:* neither good or bad! (*while thinking: I don't want to choose 'do not know' because that will show I'm clueless about the EU*)

Problem: same observed choice but different DGP behind that choice

# The ideal: a variable *in the data* that labels these two types of respondents with their corresponding DGP

| Name[2] | Choice | Label |
|---|---|---|
| Nojus | Neither good nor bad | Informed |
| Matas | Neither good nor bad | Informed |
| Vilt | Neither good nor bad | Uninformed |

[2] *Source*: Babynamewizard.com Most popular Lithuanian boys and girls names

# What we have instead: covariates that we're going to use to probabilistically model that label assignment

| Name | Choice | ~~Label~~ | Education | Age |
|---|---|---|---|---|
| Nojus | Neither good nor bad | ? | College | 45 |
| Matas | Neither good nor bad | ? | H.S. | 35 |
| Vilt | Neither good nor bad | ? | H.S. | 21 |

# The need to model who might be informed v. uninformed *before* modeling views on EU leads to a shift from one-stage to a two-stage process

Assume there is a latent variable $Y_i^*$ that in this case, represents something like the latent degree of support for Lithuana's EU membership

- **Standard ordered probit**: use covariates to model choice:
  1. 'A bad thing': low $Y_i^*$
  2. 'Neither good nor bad': medium $Y_i^*$
  3. 'A good thing': high $Y_i^*$
- That one-stage process is for 'a bad thing' and 'a good thing', but MiOP argues that middle category is likely inflated (has excess responses) because that response for observation $i$ could be generated by the following two-stage DGP:
  1. Stage one: is the respondent informed or uninformed but wants to save face?
     - If uninformed and wants to save face: chooses 'neither good nor bad'
  2. Stage two: conditional on being informed, having medium $Y_i^*$

# Putting that argument into mathematical notation: sidenote on notation we use versus notation in paper

- For consistency with Lectures 4/5, and because the authors make the *confusing* choice to use $z_i$ to refer to a vector of covariates that relates to being informed rather than a latent variable, we're shifting some notation from the paper
- In particular (and only relevant if you want to cross-ref paper eventually):
  - Modeling binary outcome of informed v. uninformed
    - Authors use: $s_i$
    - We will use: $z_i$
  - Vector of covariates that predict being informed v. uninformed
    - Authors use: $z_i$
    - We will use: $w_i$
  - Threshold parameters for ordered probit
    - Authors use: $\mu_j$
    - We use: $\psi_j$ (Lecture also sometimes uses $\tau_j$)
  - For ordered probit, they start with $j = 0$ as first category while we start with $j = 1$

# Stage one of the model: is respondent informed or uninformed (latent variable)

- Split between two sub-populations: $z_i \in \{0, 1\}$ where $0 =$ *uninformed* and $1 =$ *informed*
- Latent variable representation:
  - $z_i^*$: latent propensity to be informed
  - $w_i$: vector of covariates related to that propensity (e.g., age; whether you discuss politics)
  - $\gamma$: coefficients on those covariates
  - Putting it together: $z_i^* = w_i'\gamma + u_i$
- Translating back into binary outcomes and modeling using probit...two types of respondents, where $\Phi$ is standard normal CDF:
  1. Informed: $Pr(z_i = 1|w_i) = Pr(z_i^* > 0|w_i) = \Phi(w_i\gamma)$
  2. Uninformed: $Pr(z_i = 0|z_i) = Pr(z_i^* \leq 0|w_i) = 1 - \Phi(w_i\gamma)$

# Taking stock: we now have a model for stage one of our DGP (is the respondent informed or uninformed?)

1. Informed:
$$Pr(z_i = 1|w_i) = Pr(z_i^* > 0|w_i) = \Phi(w_i\gamma)$$
2. Uninformed:
$$Pr(z_i = 0|w_i) = Pr(z_i^* \leq 0|w_i) = 1 - \Phi(w_i\gamma)$$

How do we then incorporate this information into stage two of our DGP (views on EU?)

# Stage two: add these indicators to ordered probit model from Lecture 4, Slide 101

Where:

- $y_i$ is observed choice
- $x_i$: covariates predicting that choice - importantly, these can be different than the covariates that predict being informed or not
- $\beta$: coefs on covars
- And we generalize to $j$ choices (rather than just the $j = 3$ of EU case) where $m$ = middle choice:

$$Pr(y_i) = \begin{cases} Pr(y_i = 1|x_i, w_i) & = \Phi(w_i'\gamma)\Phi(w_i'\beta) \\ Pr(y_i = j|x_i, w_i) & = [1 - \Phi(w_i'\gamma)]^{j=m} + \Phi(w_i'\gamma)[\Phi(\psi_j - x_i'\beta) - \Phi(\psi_{j-1} - x_i'\beta)] \\ Pr(y_i = J|x_i, w_i) & = \Phi(w_i'\gamma)[1 - \Phi(\psi_{J-1} - x_i'\beta)] \end{cases}$$

# Aside about paper: correlated errors ordered probit model (MiOPC)

- Now that we've shifted from a one-stage model in the typical ordered probit case (just modeling how a respondent's covariates predict his or her choice on EU question) to a two-stage model, we run into a challenge with the error terms in each model
- More specifically, we have two one equation and one error term in each stage:
  1. Stage one: model informed v. uninformed
  $$z_i^* = w_i'\gamma + u_i$$
  2. Stage two model: choice among informed
  $$y_i^* = x_i'\beta + e_i$$
- Since both $e_i$ and $u_i$ come from the same respondent, these error terms are likely to be correlated so the authors create another model (MiOPC) that adds to the model/estimates $\rho_{eu}$

# Combining the two stages into one likelihood/log-likelihood

In writing out the likelihood, we distinguish between two cases, where $m =$ indicates middle category:

1. Observed choice is not middle category: $Pr(y_i = j|x_i, w_i)$ means we don't include observations classified as uninformed
   - **Don't include** $1 - \Phi(w_i'\gamma)$ in that part of the likelihood
2. Observed choice is middle category: $Pr(y_i = m|x_i, w_i)$
   - **Do include** $1 - \Phi(w_i'\gamma)$ in that part of the likelihood

This leads to a likelihood/log-likelihood with three components...

## Likelihood/log-likelihood for MiOP

1. $L(\gamma, \beta, \psi|x, w)^3 =$

$$\underbrace{\prod_i^n \prod_{j=1}^{m-1} [Pr(z_i = 1)Pr(y_i = j)]^{d_{ij}}}_{choose\ cat<m}$$

$$\times \underbrace{\prod_i^n \prod_{j=m}^{m} [Pr(z_i = 0) + Pr(z_i = 1)Pr(y_i = j)]^{d_{ij}}}_{choose\ cat\ m}$$

$$\times \underbrace{\prod_i^n \prod_{j>m}^{J} [Pr(z_i = 1)Pr(y_i = j)]^{d_{ij}}}_{choose\ cat>m}$$

2. $\ell(\gamma, \beta, \psi|x, w)$: $\prod$ become $\sum$ and $a^b = bln(a)$ so move $d_{ij}$ out of exponent

[3] $d_{ij}$ is an indicator for whether respondent $i$ chose category $j$

Coding this log-likelihood into R

# Coding the log-likelihood into R (from replication package)

```
MIOP <- function(b,data) {
#stores outcome
  y<-data[,1]           #EU Support

 ##stores each covar
  x1<-data[,2] #political trust
  x2<-data[,3] #xenophobia
  x3<-data[,4] #discuss politics
  x4<-data[,5] #univers_ed
  x5<-data[,6] #professional
  x6<-data[,7] #executive
  x7<-data[,8] #manual
  x8<-data[,9] #farmer
  x9<-data[,10] #unemp
  x10<-data[,11] #rural
  x11<-data[,12] #female
  x12<-data[,13] #age
  x13<-data[,16] #student
  x14<-data[,18] #income
  x15<-data[,17] #EU Bid Knowledge
  x16<-data[,14] #EU Knowledge Objective
  x17<-data[,22] #TV
  x18<-data[,23];  x19<-data[,24]; x20<-data[,25]  #High High-Mid; Low-Mid
```

## Coding the log-likelihood into R (from replication package)

```
#observations
n<-nrow(data)
#covars for stage 2 choice if informed
z<-cbind(x1,x2,x3,x5,x6,x7,x8,x9,x10,x11,x12,x13,x14,x18,x19,x20)
#covars for stage 1 - informed or not
x<-cbind(1,x3,x10,x11,x12,x13,x15,x16,x17,x18,x19,x20)

#initialize thresholds for ordered probit
tau<- rep(0,6)
tau[1]<--(Inf)
tau[2]<- b[1]
tau[3]<- b[1]+exp(b[2])
tau[4]<- (Inf)
```

# Coding the log-likelihood into R (from replication package)

```
  llik <- matrix(0, nrow=n, ncol = 1)
  #iterate through each obs
  for(i in 1:n){
  #coef for informed or not
  B<-b[3:14]; XB  <- B %*% x[i,]
  #coef for EU view
  G<-b[15:30]; ZG  <- G %*% z[i,]

  #if choice is 1 (EU bad), assume informed and estimate choice prob
 if(y[i]==1){llik[i]<-log((pnorm(XB)) * (pnorm(tau[2]-ZG)) )}

 #if choice is 2  (neither), add prob of uninformed to informed*choice
 else if(y[i]==2){llik[i]<-log((1-pnorm(XB))+
  (pnorm(XB)) * (pnorm(tau[3]-ZG) - pnorm(tau[2]-ZG)))}

 #if choice is 3, assume informed and estimate choice prob
 else if(y[i]==3){llik[i]<-log((pnorm(XB)) * (1-pnorm(tau[3]-ZG)))}
}
        llik<--1*sum(llik); return(llik)
}
```

# Estimating using one of R's built-in methods for numerical optimization

```
b<-rep(.01,30)
output.MIOP<-optim(f=MIOP, p=b, method="BFGS",
                   control=list(maxit=500),
                   data=Dataset.Expanded, hessian=TRUE)
```

*BFGS = briefly reviewed in Precept 3; 'quasi-Newton' method that takes the general form of using both the first and second derivative of the function we're max/minimizing (quasi = uses approximation for Hessian rather than analytic solution)*

# Results: first stage (predicted being informed = 1)

| | coefficient | SE | z-score |
|---|---|---|---|
| constant | 0.43 | 0.22 | 1.97 |
| discuss pol | 0.21 | 0.05 | 4.33 |
| rural | -0.09 | 0.04 | -2.29 |
| female | -0.39 | 0.08 | -4.76 |
| age | -0.01 | 0.00 | -2.98 |
| student | -0.36 | 0.16 | -2.28 |
| EU bid | 0.49 | 0.10 | 4.82 |
| EU_know | 0.15 | 0.02 | 6.94 |
| TV | 0.06 | 0.03 | 1.97 |
| high | -0.22 | 0.14 | -1.62 |
| high-mid | -0.52 | 0.14 | -3.81 |
| low-mid | -0.48 | 0.09 | -5.22 |

# Results: second stage (conditional on informed, choice of category)

| | coefficient | SE | z-score |
|---|---|---|---|
| polit_trust | 0.90 | 0.05 | 17.43 |
| Xenophobia | -0.58 | 0.05 | -10.94 |
| discuss_politics | 0.02 | 0.02 | 0.81 |
| professional | -0.09 | 0.08 | -1.15 |
| executive | 0.12 | 0.10 | 1.19 |
| manual | -0.13 | 0.05 | -2.71 |
| farmer | -0.05 | 0.09 | -0.56 |
| Unemployed | 0.12 | 0.06 | 2.10 |
| rural | 0.01 | 0.02 | 0.44 |
| female | 0.03 | 0.03 | 0.81 |
| age | -0.00 | 0.00 | -1.45 |
| student | 0.15 | 0.08 | 1.78 |
| income | 0.07 | 0.01 | 10.75 |
| high | 0.09 | 0.06 | 1.52 |
| high-mid | 0.01 | 0.07 | 0.13 |
| low-mid | -0.03 | 0.05 | -0.72 |

## Takeaways

- Same observed choice–neither good nor bad–in population of respondents disguises two sub-populations: informed about EU bid and genuinely torn versus uninformed and saving face
- Ideal data: we'd have a label identifying each observation as belonging to one of those two sub-populations
- Real data: we lack that label so model it using covariates
- Adding that modeling of the label transforms a typical ordered probit case into a two-stage process, just as in the zero-inflation models, adding in the logit/probit as the first stage transforms a typical count model into a two-stage process

# Transition to EM

- In previous optimization, we ended up with a challenging log-likelihood involving a normal pdf that required using a *numerical optimization* method (BFGS)
- EM is focused on similarly intractable likelihoods that have two characteristics:
    1. We can write the model using a latent variable representation, which we can do with mixture models
    2. When we get to a step reviewed on a later slide, we are able to take the expectation to compute responsibilities

# Motivating example: classification of wine based on observed attributes

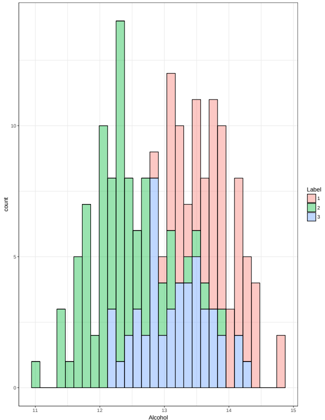
- Opening up a script or .rmd, install and load the `gclus` package
- Then, load and view the `wine` data

- You'll notice there's a variable `Class` that indicates which cultivar out of $k = 3$ options the wine comes from
- When coding the EM algorithm, we'll use that to check our work
- But the example is motivated by idea that those labels are *latent variables* that we need to probabilistically estimate

To estimate those labels, we'll return to the normal/Gaussian mixture we've seen in many examples...

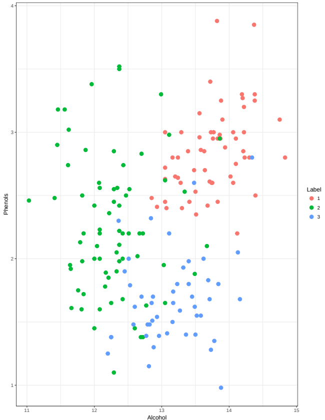
# What we did with student heights and old faithful...univariate normal

- In univariate normal, we model a given wine's label using a *single observed attribute* among the many present in the data (e.g., choose one out of Alcohol; Phenols; Ash, etc)



# But we might think that if we add in other attributes, we can better distinguish between labels/clusters

- Rather than estimate labels based on a single attribute (Alcohol content), can estimate labels based on multiple attributes (in this case: Alcohol content + Phenols)



Conveniently, this takes us into the world of the EM algorithm derivation from Slides 38 and 39 of this week's lecture...we're going to go slowly step by step and implement in `R` using the wine data, Alcohol and Phenol attributes (bivariate normal), and $k = 3$

# Algorithm for Gaussian mixture

1) Initialize parameters $\boldsymbol{\mu}^t, \boldsymbol{\Sigma}^t$, $\boldsymbol{\pi}^t$
2) Expectation step: compute 'responsibilities' $p(\boldsymbol{z}_i|\boldsymbol{\mu}^t, \boldsymbol{\Sigma}^t, \boldsymbol{\pi}^t, \boldsymbol{X}) \rightsquigarrow \boldsymbol{r}_i^t$

$$r_{ik} = \frac{\pi_k \mathcal{N}(x_i|\mu_k, \Sigma_k)}{\sum_{k'} \pi_{k'} \mathcal{N}(x_i|\mu_{k'}, \Sigma_{k'})}$$

3) Maximization step: maximize with respect to $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ and $\boldsymbol{\pi}$:

$$\begin{aligned} \mathrm{E}_z[\log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\pi})] &= \mathrm{E}_z\left[\log\left(\prod_{i=1}^{N}\prod_{k=1}^{K} \pi_k^{z_{nk}} \mathcal{N}(x_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{nk}}\right)\right] \\ &= \mathrm{E}_z\left[\sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}\left[\log \pi_k \mathcal{N}(x_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]\right] \end{aligned}$$

Obtain $\boldsymbol{\mu}_k^{t+1}, \boldsymbol{\Sigma}_k^{t+1}$, $\boldsymbol{\pi}^{t+1}$

4) Assess change in the log-likelihood

# Focus on M-step

3) M-Step:

$$\text{E}[\log \text{Complete data}|\boldsymbol{\theta}, \boldsymbol{\pi}] = \sum_{i=1}^{N} \sum_{k=1}^{K} E[z_{ik}] \log\left(\pi_k \mathcal{N}(x_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right)$$

Because $E[z_{ik}] = r_{ik}$, solutions are weighted averages of usual updates

$$\pi_k^{t+1} = \frac{\sum_{i=1}^{N} r_{ik}^t}{N} \tag{1}$$

$$\mu_k^{t+1} = \frac{\sum_{i=1}^{N} r_{ik}^t x_i}{\sum_{i=1}^{N} r_{ik}^t} \tag{2}$$

$$\Sigma_k^{t+1} = \frac{1}{\sum_{i=1}^{N} r_{ik}^t} \sum_{i=1}^{N} r_{ik} (x_i - \boldsymbol{\mu}_k^{t+1})(x_i - \boldsymbol{\mu}_k^{t+1})^T \tag{3}$$