Classification models

Filipe Rodrigues

Francisco Pereira

DTU Management Engineering
Department of Management Engineering

Outline

- Case study: Modeling travel mode choices
- Logistic regression
- Generalized linear models (GLMs)
- Hierarchical models

Learning objectives

At the end of this lecture, you should be able to:

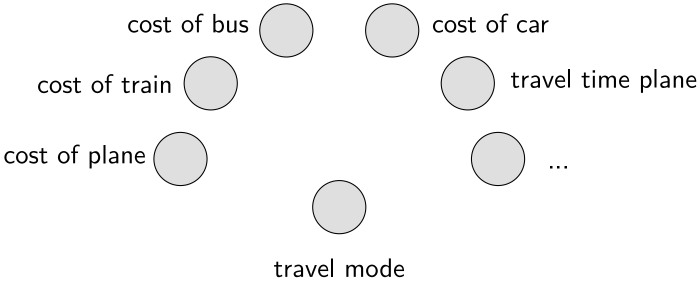
- Explain what (Bayesian) logistic regression is and its underlying assumptions
- Relate different Generalized Linear Models (GLMs)
- Explain the extension of logistic regression to multiple classes
- Explain the underlying concepts and assumptions behind hierarchical models
- Relate different ways of modelling the dependency of a discrete random variable on other variables and justify their suitability for a problem
- Implement the modelling techniques above in STAN/Pyro

Modeling travel mode choices

- Travel diary data
  - 394 survey observations from 80 individuals
  - 4 travel modes: plane, train, bus or car
- Goal: **model user mode choices**
- Trip attributes (features):
  - Terminal waiting time
  - Cost (dollars)
  - Travel time (minutes)
  - Household income
  - Traveling group size
- Some possible applications:
  - Understanding people's choices
  - Developing pricing policies
  - Incentivising mode change
  - Suggesting car pooling

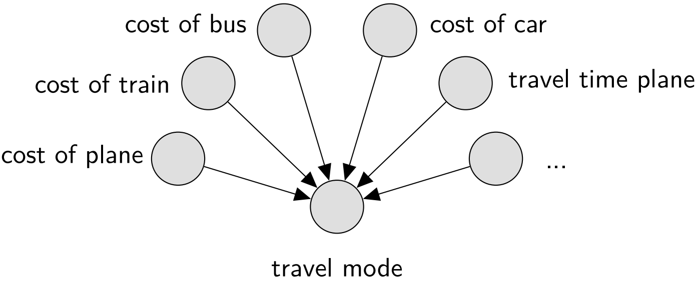
Modeling travel mode choices (cont'd)

- Let's start thinking about the graphical model...



Modeling travel mode choices (cont'd)

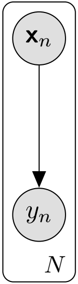
- Let's start thinking about the graphical model...



- What distribution should we assign to the "travel mode" variable?
  - Travel mode is a **discrete variable**!
  - We are now in a **classification** setting
- How should we model the dependency of the travel mode on the other variables?

Discrete output variables

- We can represent our model for the entire dataset compactly as:



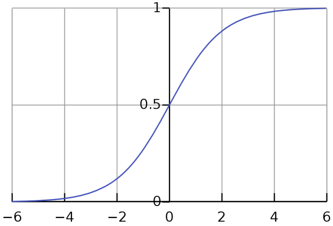
N is the number of trips in the dataset
y_n is the travel mode of the n^{th} trip in the dataset
$\mathbf{x}_n$ is a vector with $\{$cost of plane, cost of train, $\ldots\}$ for trip n

- Looks familiar?
- But how should we model the dependency of y_n on $\mathbf{x}_n$?
  - We can assume a parameterized linear relationship: $y_n = \boldsymbol{\beta}^{\mathsf{T}}\mathbf{x}_n$
  - But $y_n \notin \mathbb{R}$! Instead: $y_n \in \{$plane, train, bus, car$\}$

Binary logistic regression

- Consider the binary case: $y_n \in \{0, 1\}$
- We need a function that maps from $\mathbb{R}$ to $[0, 1]$
- A sigmoid ("S"-shaped) function does precisely that!
- E.g. logistic sigmoid:

$$\text{Sigmoid}(z) = \frac{1}{1 + e^{-z}} = \frac{e^z}{e^z + 1}$$

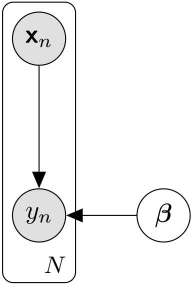


- We can define $z_n = \boldsymbol{\beta}^\mathsf{T} \mathbf{x}_n$
- The value of $\text{Sigmoid}(z_n)$ can then be interpreted as the probability of the n^{th} instance belonging to class "1": $p(y_n = 1)$
- The probability of class "0" is simply: $p(y_n = 0) = 1 - \text{Sigmoid}(z_n)$

Binary logistic regression as a graphical model

- We have a dataset $\mathcal{D}$ consisting of N observations of the targets $y_n \in \{0, 1\}$ which depend on their corresponding explanatory variables $\mathbf{x}_n$

$$\mathcal{D} = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$$



- Generative process
  1. Draw coefficients $\beta \sim \mathcal{N}(\beta|\mathbf{0}, \lambda\mathbf{I})$
  2. For each feature vector $\mathbf{x}_n$
     a. Draw class $y_n \sim \text{Bernoulli}(y_n|\text{Sigmoid}(\beta^{\mathsf{T}}\mathbf{x}_n))$

- Joint probability distribution factorizes as

$$p(\mathbf{y}, \beta|\mathbf{X}, \lambda) = \underbrace{p(\beta|\lambda)}_{\text{prior}} \times \underbrace{\prod_{n=1}^{N} p(y_n|\beta, \mathbf{x}_n)}_{\text{likelihood}}$$

where $\mathbf{y} = \{y_n\}_{n=1}^{N}$, $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^{N}$ and β are the model parameters.

Multi-class logistic regression

- What if we have multiple classes? (like in our mode choice example...)

$$y_n \in \{\text{plane, train, bus, car}\}$$

- The generalization of the logistic sigmoid to multiple outputs is the **softmax**:

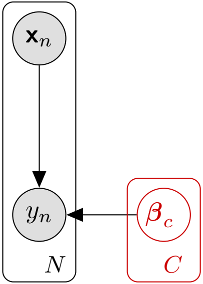
$$\text{Softmax}(\mathbf{x}_n, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C)_c = \frac{\exp(\boldsymbol{\beta}_c^\mathsf{T} \mathbf{x}_n)}{\sum_{k=1}^{C} \exp(\boldsymbol{\beta}_k^\mathsf{T} \mathbf{x}_n)}, \quad \text{for } c \in \{1, \dots, C\}$$

  where C denotes the number of classes

- Notice that we now need C vectors of parameters: $\{\boldsymbol{\beta}_1, \dots \boldsymbol{\beta}_C\}$
- The output of the softmax is then a vector $\boldsymbol{\eta} = [\eta_1, \dots, \eta_C]$ where $\eta_c = \text{Softmax}(\mathbf{x}_n, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C)_c$
- The value of η_c can be interpreted as the probability of the n^{th} instance belonging to class c
- The softmax ensures that $\sum_{c=1}^{C} \eta_c = 1$

Multi-class logistic regression as a graphical model

- Updated graphical model



- Generative process

1. For each class $c \in \{1, \dots, C\}$
   a. Draw coefficients $\boldsymbol{\beta}_c \sim \mathcal{N}(\boldsymbol{\beta}_c | \mathbf{0}, \lambda \mathbf{I})$
2. For each feature vector $\mathbf{x}_n$
   a. Draw class $y_n \sim \text{Multinomial}(y_n | \text{Softmax}(\mathbf{x}_n, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C))$

- Joint probability distribution factorizes as

$$p(\mathbf{y}, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C | \mathbf{X}, \lambda) = \underbrace{\left(\prod_{c=1}^{C} p(\boldsymbol{\beta}_c | \lambda) \right)}_{\text{prior}} \times \underbrace{\prod_{n=1}^{N} p(y_n | \mathbf{x}_n, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C)}_{\text{likelihood}}$$

Inference

- **Goal:** compute **posterior** distribution on $\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C$
- Following Bayes' theorem

$$\underbrace{p(\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C | \mathbf{y}, \mathbf{X}, \lambda)}_{\text{posterior}} \propto \underbrace{\left(\prod_{c=1}^{C} \mathcal{N}(\boldsymbol{\beta}_c | \mathbf{0}, \lambda \mathbf{I}) \right)}_{\text{prior}}$$
$$\times \underbrace{\prod_{n=1}^{N} \text{Multinomial}(y_n | \text{Softmax}(\mathbf{x}_n, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_C))}_{\text{likelihood}}$$

- Exact inference is intractable
- Must resort to **approximate inference** methods
- Not a problem for Stan :-)

Playtime!

- Ancestral sampling from multi-class logistic regression model
  - See “Logistic regression - Ancestral sampling.ipynb" notebook
  - Expected duration: 15 minutes
- Bayesian multi-class logistic regression model of travel mode choices
  - See “Travel mode choice - Logistic regression.ipynb" notebook
  - Expected duration: 1 hour

Generalized linear models (GLMs)

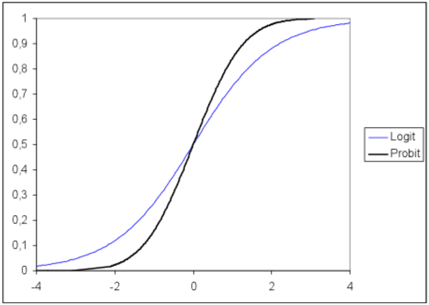
- So far we saw a series of linear models
  - Linear regression
  - Poisson regression
  - Logistic regression
- The parameters $\boldsymbol{\beta}$ enter the distribution of y_n through a linear combination of $\mathbf{x}_n$
- The difference is in the distribution of the response
  - Gaussian for linear regression
  - Poisson for poisson regression
  - Bernoulli for binary logistic regression
  - Multinomial for multi-class logistic regression
- In other words, we just changed the **form of the likelihood**!
- All belong to a general class of models called **generalized linear models**
  - The idea is to use a general exponential family for the response distribution
  - Can handle real, binary, categorical, positive real, positive integer and ordinal responses

Probit regression

- Another example of a generalized linear model
- Very similar to logistic regression
- But uses a **different link function**: probit instead of the logistic sigmoid
- Probit function Φ is the CDF of the standard Gaussian distribution $\mathcal{N}(0, 1)$

$$\Phi(z) = \int_{-\infty}^{z} \mathcal{N}(t|0,1)\,dt$$
$$= \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)\right]$$

where $\mathrm{erf}(\cdot)$ is a special function



- Generative process

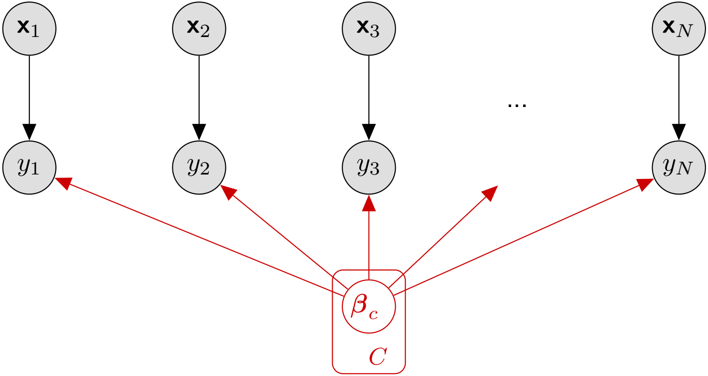
1. Draw coefficients $\beta \sim \mathcal{N}(\beta|\mathbf{0}, \lambda\mathbf{I})$
2. For each feature vector $\mathbf{x}_n$
   a. Draw class $y_n \sim \mathrm{Bernoulli}(y_n|\Phi(\beta^\mathsf{T}\mathbf{x}_n))$

Playtime!

- Probit regression vs logistic regression model
- See “Travel mode choice - Probit regression.ipynb" notebook

Going back to our travel mode choice case study...

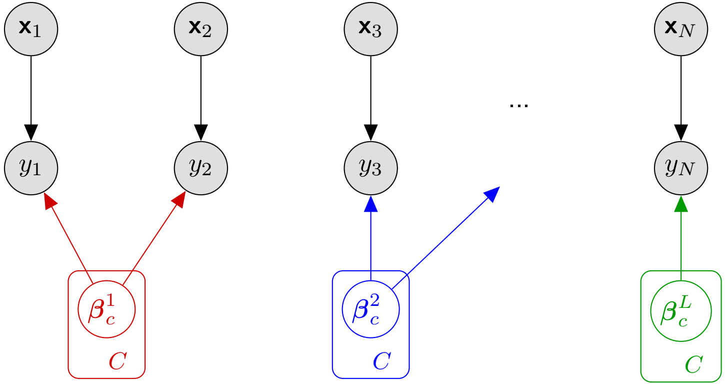
- Let's revise the **modeling assumptions** that we made



- Single set of parameters $\{\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_C\}$ for **all** the observations
  - This corresponds to saying that all individuals give the same importance (weight) to all the features (e.g. travel time) and have the same biases!

Going back to our travel mode choice case study...

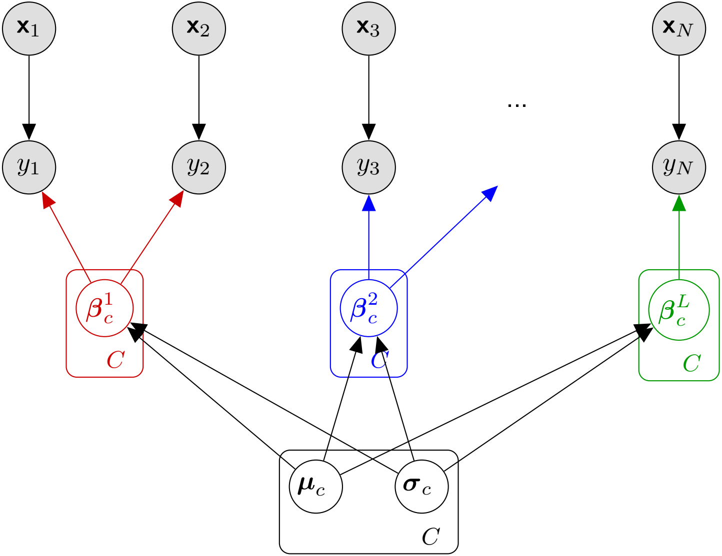
- Alternatively, we can assign each individual his/her own parameters



- Each individual $l \in \{1, \ldots, L\}$ gets his/her own set of parameters $\{\beta_1^l, \ldots, \beta_C^l\}$
  - Allows to capture personalized preferences and biases
  - But can lead to terrible overfitting! (more parameters than observations)

Going back to our travel mode choice case study...

- A compromise between the two: **hierarchical models**



Hierarchical models

- Assume the data is grouped into L distinct levels (or groups)
  - In our travel mode choice example, levels correspond to e.g. individuals
- Data from each level l gets its own set of parameters
- Shared global prior ("**hyper-prior**") ties together the parameters of each level l
- A compromise between two extremes:
  - On one extreme, each level l gets its own set of parameters (no pooling)
  - On the other extreme, all the observations share a single set of parameters (complete pooling)
- The degree of pooling is determined by the data and the specified priors

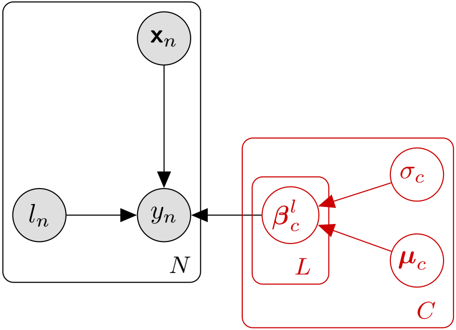
Note

This concept can also be applied to other types of models! E.g. linear regression, poisson regression, etc.

Hierarchical logistic regression model

- Probabilistic graphical model

- l_n is used to denote the level (or group) that the n^{th} observation belongs to



- Joint probability distribution:

$$p(\mathbf{y}, \mathbf{B}^1, \dots, \mathbf{B}^L, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_C, \sigma_1, \dots, \sigma_C | \mathbf{X}, \mathbf{l})$$

$$= \underbrace{\left(\prod_{c=1}^{C} p(\boldsymbol{\mu}_c)\, p(\sigma_c) \prod_{l=1}^{L} p(\boldsymbol{\beta}_c^l | \boldsymbol{\mu}_c, \sigma_c) \right)}_{\text{hierarchical prior}} \times \underbrace{\prod_{n=1}^{N} p(y_n | \mathbf{x}_n, l_n, \mathbf{B}^1, \dots, \mathbf{B}^L)}_{\text{likelihood}}$$

where we defined $\mathbf{B}^l = \{\beta_1^l, \dots, \beta_C^l\}$

Hierarchical logistic regression model

- Generative process

1. For each class $c \in \{1, \dots, C\}$
   a. Draw global mean parameters $\boldsymbol{\mu}_c \sim \mathcal{N}(\boldsymbol{\mu}_c | \mathbf{0}, \lambda \mathbf{I})$
   b. Draw global variance parameter $\sigma_c \sim \mathcal{N}(\sigma_c | 0, \tau)$
   c. For each level $l \in \{1, \dots, L\}$
      a. Draw coefficients $\boldsymbol{\beta}_c^l \sim \mathcal{N}(\boldsymbol{\beta}_c^l | \boldsymbol{\mu}_c, e^{\sigma_c} \mathbf{I})$
2. For each feature vector $\mathbf{x}_n$
   a. Draw class $y_n \sim \text{Multinomial}(y_n | \text{Softmax}(\mathbf{x}_n, \boldsymbol{\beta}_1^{l_n}, \dots, \boldsymbol{\beta}_C^{l_n}))$

- There are many variants of this that we can consider
  - A vector of variances $\boldsymbol{\sigma}_c$ rather than a single variance σ_c for all the features
  - Different prior distributions on $\boldsymbol{\mu}_c$, σ_c and even $\boldsymbol{\beta}_c^l$
  - Hierarchical prior only on the biases (intercepts) rather than on all the $\boldsymbol{\beta}_c$
  - More levels, etc.

Playtime!

- Bayesian **hierarchical** multi-class logistic regression model of travel mode choices
- Each individual has his/her own bias towards certain travel modes
- See “Travel mode choice - Hierarchical models.ipynb" notebook
- Expected duration: 1 hour