Sequence labelling. Ordered outcomes classification.

Victor Kitov

v.v.kitov@yandex.ru

Table of Contents

1. Sequence labelling
2. Ordered outcomes classification.

Collective classification

Collective classification:

for set of $x_1, ...x_N$ find corresponding $y_1, ...y_N$ jointly.

- Appears mostly in graphs
- Example: for each person in social graph predict his education
  - close people on the graph have usually similar background
  - classifications affect each other
  - need to classify all people simulataneously
- Particular case - sequence labelling

Sequence labelling

Sequence labelling:

Assign $x_1 \ldots x_N$ labels $y_1, \ldots y_N$ where neighbouring labels are dependent.

Applications of sequence labelling:

- Part-of-speech tagging
  - close parts-of-speech are dependent
- Speech recognition
  - close sounds/words are dependent
- Handwriting recognition
  - close letters/words are dependent

Sequence prediction Markov model

- Sequence prediction Markov model prediction

$$\widehat{Y} = \arg\max_{Y} p(Y|X) = \arg\max_{Y} \prod_{n=1}^{N} p(y_n|x_n, y_{n-1})$$

- Graphical structure:

$$\begin{tikzcd} y_1 \arrow[r] & y_2 \arrow[r] & \cdots \arrow[r] & y_N \\ x_1 \arrow[u] & x_2 \arrow[u] & \cdots & x_N \arrow[u] \end{tikzcd}$$

Naive prediction

- For simplicity consider conditioning y_n only on X and y_{n-1}.
- Naive prediction:

```
for n = 1, 2, ...N:
    y_n = arg max_y p(y|y_{n-1}, X)
```

  - fast
  - makes greedy, local decisions
  - cannot correct earlier decisions from later inconsistencies
- Viterbi algorithm gives a consistent sequence of predictions for whole sequence.

Viterbi algorithm: forward pass[1]

Assume $p(y_t|history) = p(y_t|x_t, y_{t-1})$. Definitions:

$$\varepsilon_t(i, X) := \max_{y_{1,\dots} y_{t-1},} p\left(y_1 \dots y_{t-1} y_t = i | x \dots x_t\right)$$

$$v_t(i, X) := \arg\max_j p(y_1 \dots y_{t-2}, y_{t-1} = j, y_t = i | x_1 \dots x_t)$$

Init:

$$\varepsilon_1(i, X) = p(y_1 = i|x_1) = \text{output of classifier}$$

For $t = 1, \dots T-1$:

$$\begin{aligned}
\varepsilon_{t+1}(i, X) &= \max_{y_1 \dots y_{t-1}, j} p(y_1 \dots y_{t-1} y_t = j, y_{t+1} = i | x_1 \dots x_t x_{t+1}) \\
&= \max_j \max_{y_1 \dots y_{t-1}} p(y_1 \dots y_{t-1} y_t = j | x_1 \dots x_{t+1}) p(y_{t+1} = i | y_1 \dots y_{t-1} y_t = j, x_1 \dots x_{t+1}) \\
&= \max_j \max_{y_1 \dots y_{t-1}} p(y_1 \dots y_{t-1} y_t = j | x_1 \dots x_t) p(y_{t+1} = i | y_t = j, x_{t+1}) \\
&= \max_j \varepsilon_t(j, X) p(y_{t+1} = i | y_t = j, x_{t+1})
\end{aligned}$$

$$v_{t+1}(i, X) = \arg\max_j \varepsilon_t(j, X) a_{ji}$$

---

[1] Propose algorithm modification for looking at 2 previous predictions.

Viterbi algorithm: backward pass

Definitions

$$y_1^*, ...y_T^* := \arg\max_{y_1,...y_T} p(y_1, ...y_T | x_1, ...x_T)$$

$$\varepsilon_t(i, X) := \max_{y_1,...y_{t-1},} p\left(y_1...y_{t-1} y_t = i | x_1...x_t\right)$$

$$v_t(i, X) := \arg\max_j p(y_1...y_{t-2}, y_{t-1} = j | y_t = i, x_1...x_t)$$

Init:

$$p^*(X) = \max_j \varepsilon(j, X)$$

$$y_T^*(X) = \arg\max_j \varepsilon(j, X)$$

For $t = T-1, T-2, ...1$:

$$y_t^*(X) = v_{t+1}(y_{t+1}^*(X))$$

Comments

- We could define $\varepsilon_t(i,X) := \max_{y_1,...y_{t-1}} p\left(y_1...y_{t-1}y_t = i|x...x_t x_{t+1}...x_{t+k}\right)$ for some lookahead horizon $k>0$.
- we could condition y_t on several states before $y_{t-1}, y_{t-2}, ...$
- Instead of left-to-right classification we could use use right-to-left classification
    - and then combine their outputs in an ensemble
- Also we could make several passes:
    - first pass: obtain most likely $\widehat{y}_1^1, ...\widehat{y}_N^1$
    - second pass: make classification both on past *and future:*

$$p(y_t|context(t)) = p(y_t|x_t \widehat{y}_{t-k}^2 ... \widehat{y}_{t-1}^2 \widehat{y}_t^1 \widehat{y}_{t+1}^1 ... \widehat{y}_{t+k}^1)$$

$$\widehat{Y} = \arg\max_Y p(Y|X) = \arg\max_Y \prod_{n=1}^{N} p(y_n|context(t))$$

Table of Contents

1. Sequence labelling
2. Ordered outcomes classification.

Problem statement

- Suppose $y \in \{\omega_1, ...\omega_C\}$ and classes can be ordered.
- Examples:
    - for people with different characteristics need to predict their income
        - $income \in \{low, medium, high\}$, $low < medium < high$
    - travellers select place of vacation
        - $place \in \{dacha, Crimea, Greece\}$, $dacha < Crimea < Greece$
- Accounting for order increases prediction accuracy.

Logit model

- Logit model:
  - $z_n = x_n^T w + \varepsilon_n$
  - $\varepsilon_n \sim F(u) = Logistic(u) = \frac{1}{1+e^{-u}}$
  - $y_n = \begin{cases} +1, & z_n \geq 0 \\ -1, & z_n < 0 \end{cases}$

$$p(y_n = 1) = p(z_n \geq 0) = p(x_n^T w + \varepsilon_n \geq 0) = p(\varepsilon_n \geq -x_n^T w)$$
$$= p(\varepsilon_n < x_n^T w) = Logistic(x_n^T w) = \frac{1}{1 + e^{-x_n^T w}}$$

- Logit model=logistic regression!

Ordered logit model - 3 outcomes

- Ordered logit model for 3 outcomes:
  - $z_n = x_n^T w + \varepsilon_n$
  - $\varepsilon_n \sim F(u) = Logistic(u) = \frac{1}{1+e^{-u}}$
  - $y_n = \begin{cases} 1, & z_n \le c_1 \\ 2, & c_1 < z_n \le c_2 \\ 3, & z_n > c_2 \end{cases}$

$$
\begin{aligned}
p(z_n \le c_1) &= p(\varepsilon_1 \le c_1 - x_n^T w) = F(c_1 - x_n^T w) \\
p(c_1 < z_n \le c_2) &= p(c_1 - x_n^T w \le \varepsilon_1 \le c_2 - x_n^T w) \\
&= F(c_2 - x_n^T w) - F(c_1 - x_n^T w) \\
p(z_n > c_2) &= p(\varepsilon_2 > c_2 - x_n^T w) = 1 - F(c_2 - x_n^T w)
\end{aligned}
$$

Optimization

- Optimization:

$$\prod_{n:y_n=1} F(c_1 - x_n^T w) \prod_{n:y_n=2} \left(F(c_2 - x_n^T w) - F(c_1 - x_n^T w)\right) \times$$
$$\times \prod_{n:y_n=3} \left(1 - F(c_2 - x_n^T w)\right) \to \max_{w, c_1, c_2, c_3}$$

Ordered logit model - m outcomes

- Ordered logit model for m outcomes:
  - $z_n = x_n^T w + \varepsilon_n$
  - $\varepsilon_n \sim F(u) = Logistic(u) = \frac{1}{1+e^{-u}}$
  - $-\infty = c_0 < c_1 < ... < c_{m-1} < c_m = \infty$
  - $y_n = j \Leftrightarrow c_{j-1} < z_n \leq c_j$
- Probability of outcome:

$$
\begin{aligned}
p(y_n = j) &= p(c_{j-1} - x_n^T w < \varepsilon_n \leq c_j - x_n^T w) \\
&= F(c_j - x_n^T w) - F(c_j - x_n^T w)
\end{aligned}
$$

- Optimization:

$$
\prod_{j=1}^{m} \prod_{n:y_n=j} \left(F(c_j - x_n^T w) - F(c_j - x_n^T w) \right) \to \max_{w, c_1, \dots c_{m-1}}
$$

Comments

- Application example: how bank ratings depend on their financial parameters?
- OrderedLogit implicitly uses here that its better to misclassify LOW as MED than LOW as HIGH
  - can have similar effect with special cost matrix (costs increase as we move off the diagonal)
  - but ordered logit imposes **continously increasing cost for errors in real-valued score**.
- Generalizations:
  - $z_n = f(x_n, w) + \varepsilon_n$
  - $\varepsilon_n \sim F(u)$, $F(u)$-not logistic.
    - if $F(u)$-standard normal, model is called **probit**.

Censored regression[2]

- Censored regression (Tobit model):
  - $y_n = \begin{cases} \gamma, & x_n^T w < \gamma \\ x_n^T w, & x_n^T w \geq \gamma \end{cases}$
  - $y_n = \begin{cases} \gamma_1, & x_n^T w < \gamma_1 \\ x_n^T w, & x_n^T w \in [\gamma_1, \gamma_2] \\ \gamma_2, & x_n^T w > \gamma_2 \end{cases}$
- Similar estimation approach may be used
- Application example: measurements in limited scale

[2] Write out optimization problem.