Topic Models with Me
——ペンギンでも分かるトピックモデル——

@anemptyarchive[*]

2020/01/28-2020/02/05

Contents

はじめに

楽しい！…よね？

[*]https://www.anarchive-beta.com/

1　文書生成

1.1　テキストの生成過程

とあるミックスジュースの紹介文 ···

> 「ももとりんごとレモンとぶどうとメロンとオレンジの搾りたて 100% です。甘酸っぱいぶどうをたっぷり使っています！あまーいもも、ぶどう、メロンを酔いしれちゃえ〜」

··· がありました。このテキストを例として扱うことにします。

このテキストを単語ごとに切り分け、各単語の出現回数を表にします。ただし、テキストマイニングにおいては、意味を持たない助詞等はあまり重要ではありません。そこで、名詞のみを取り上げることにします。

観測された単語とその頻度を五十音順に並べると次のようになります。

単語	出現回数
オレンジ	1
ぶどう	3
メロン	2
もも	2
りんご	1
レモン	1

これを BoW(bag-of-words) と呼びます。単語間の繋がりはなく、文書中に出現した単語の種類と頻度にのみ注目します。

この頻度表 (観測データ) から、そもそもどの単語がどれくらい出やすいのだろうか? その尤もらしい確率を推定していくことが目的の 1 つです。もう 1 つの目的は、どのようなトピックから文書が生成されているのかです。

トピックモデルでは、単語は確率的に生成されると仮定します。確率的にとは、要は各面に単語が書いてあるサイコロを振ってその出た目が出現した単語に相当するということです。サイコロの出目のように、各単語は前後の影響を受けず独立に生成されます。BoW として、文章としての連なりがない、とはこういうことです。

では、確率的に生成される過程を見ていきましょう。

1.2　単語の生成確率

単語を w を使って示すことにします[1]。また各単語を出現順にナンバリングして、それぞれを

n	$\boldsymbol{w}$	単語
1	w_1	もも
2	w_2	りんご
3	w_3	レモン
4	w_4	ぶどう

[1]単語 (word) の w です。

n	$\boldsymbol{w}$	単語
5	w_5	メロン
6	w_6	オレンジ
7	w_7	ぶどう
8	w_8	もも
9	w_9	ぶどう
10	w_{10}	メロン

と表現することにします。
更に、単語 w_n の集まりを

$$\boldsymbol{w} = (w_1, w_2, \cdots, w_{10})$$

と太字を使って表記することにします。この単語の集まりを単語集合 $\boldsymbol{w}$ と呼び、文書 (テキスト) のことです。文書に含まれる総単語数を N とします。つまり、添字の n は 1 から N までの値をとります。このことを

$$n = \{1, 2, \cdots, N\}$$

と表記します。この例では $N = 10$ です。

トピックモデルでは、単語はサイコロを振るように確率的に生成されるとします (そういう世界を想定しています)。これは、「もも」という単語の次には「甘い」や「美味しい」が来るだろうという文脈的な要素を考慮しないということです。そういった要素はあくまで出現のしやすさ (出現確率) として単語分布で表現されています。
例文の 6 種類の単語を使った 9 語からなるテキストは、目が「オレンジ」「ぶどう」「メロン」「もも」「りんご」「レモン」のサイコロを 9 回振り、その出目からなる。というイメージです。
「もも」という単語 w_1 が出現する確率を $p(w_1)$ と表記します。各単語は互いに影響することなく出現するので、独立した事象の同時確率で表現します。

$$\begin{aligned} p(\boldsymbol{w}) &= p(w_1, w_2, \cdots, w_{10}) \\ &= p(w_1) * p(w_2) * \cdots * p(w_{10}) \\ &= \prod_{n=1}^{10} p(w_n) \end{aligned}$$

サイコロの目は重複しません。重複せずに表現する単語を語彙と呼ぶことにして、文書中に出現する単語と呼び分けることにします。各語彙は v を添字に用いることにします[2]。

$$v = \{1, 2, \cdots, V\}$$

例文では $V = 6$ ですね。
単語の出現確率を ϕ を使って示します[3]。ここでは、「オレンジ」「ぶどう」「メロン」「もも」「りんご」「レモン」の 50 音順に 1 から 6 としておきましょう。つまり、「オレンジ」の出現確率 p(オレンジ) は ϕ_1 です。

v	単語	生成確率	出現回数
1	オレンジ	ϕ_1	1

[2]語彙 (vocabulary) の v です。
[3]確率 (probability) の p に相当するギリシャ文字の ϕ です。

v	単語	生成確率	出現回数
2	ぶどう	ϕ_2	3
3	メロン	ϕ_3	2
4	もも	ϕ_4	2
5	りんご	ϕ_5	1
6	レモン	ϕ_6	1

それぞれの生成確率を足し合わせると 1 になります。

$$\sum_{v=1}^{6} \phi_v = 1$$

総語彙数を V とすることにします (つまりこの例では $V = 6$)。各語彙の出現確率 ϕ_v もまとめて太字で表現することにします。

$$\boldsymbol{\phi} = (\phi_1, \phi_2, \cdots, \phi_V)$$

$\boldsymbol{\phi}$ を単語分布と呼びます。
また、語彙 v の出現回数を N_v とすると、総語彙数 N は

$$\sum_{v=1}^{V} N_v = N$$

とします。例文だと

$$\begin{aligned}\sum_{v=1}^{6} N_v &= N_1 + N_2 + \cdots, N_V \\ &= 1 + 3 + 2 + 1 + 1 + 1 \\ &= 9 = N\end{aligned}$$

となります。
単語の出現回数を語彙の出現回数に置き換えると、このテキストが生成される確率 (このテキストに含まれる単語の同時確率) は

$$\begin{aligned}p(\boldsymbol{w}) &= p(w_1) * p(w_2) * \cdots * p(w_{10}) \\ &= \phi_4 * \phi_5 * \phi_6 * \phi_2 * \phi_3 * \phi_1 * \phi_2 * \phi_4 * \phi_2 * \phi_3 \\ &= \phi_1 * \phi_2^3 * \phi_3^2 * \phi_4^2 * \phi_5 * \phi_6 \\ &= \prod_{v=1}^{6} \phi_v^{N_v}\end{aligned}$$

このように表現できます。指数部分は各語彙の出現回数になります。「ぶどう」が 3 回出現する確率は、$p(\text{ぶどう}) = \phi_2$ を 3 回掛けることになります。

単語分布 $\boldsymbol{\phi}$ に従って単語集合 $\boldsymbol{w}$ が生成されることを、条件付き確率 $p(\boldsymbol{w}|\boldsymbol{\phi})$ で表現することにします。
するとある単語集合 (文書)$\boldsymbol{w}$ の生成確率は

$$p(\boldsymbol{w}|\boldsymbol{\phi}) = \prod_{v=1}^{V} \phi_v^{N_v}$$

と表記できます。

1.3 R でやってみる

tidyverse パッケージは、掲載しているほぼ全てのプログラムで利用しています。今後は特に記載がなくとも読み込んでください。

```
library(tidyverse)
```

まずは歪んだ (あるいは重心のズレた) サイコロを設定します。

```
# サイコロを設定する
dice <- tibble(
  v_index = c("オレンジ", "ぶどう", "メロン", "もも", "りんご", "レモン"),
  phi_v = c(1/6, 1/6, 1/6, 1/6, 1/6, 1/6)
)

# 語彙数の設定
V <- nrow(dice)
```

6 面のサイコロである必要はありません。任意の数の語彙と対応する確率を指定してください。ここでは例のテキストより 6 語指定しています。
行数が語彙数 V に相当します。

作成した dice を使って単語を生成します。

```
# 単語数（サイコロを振る回数）を設定する
N <- 100

# サイコロを振る
w_n <- sample(dice$v_index, size = N, replace = TRUE, prob = dice$phi_v)
w_n[12]
```

```
## [1] "メロン"
```

生成する単語数 (試行回数)N を指定します。
sample() の第 1 引数に語彙、size に N(単語数)、prob に dice$phi_v(単語分布) を指定します。

```
# 出目を集計する
N_v_table <- table(w_n)
N_v_table
```

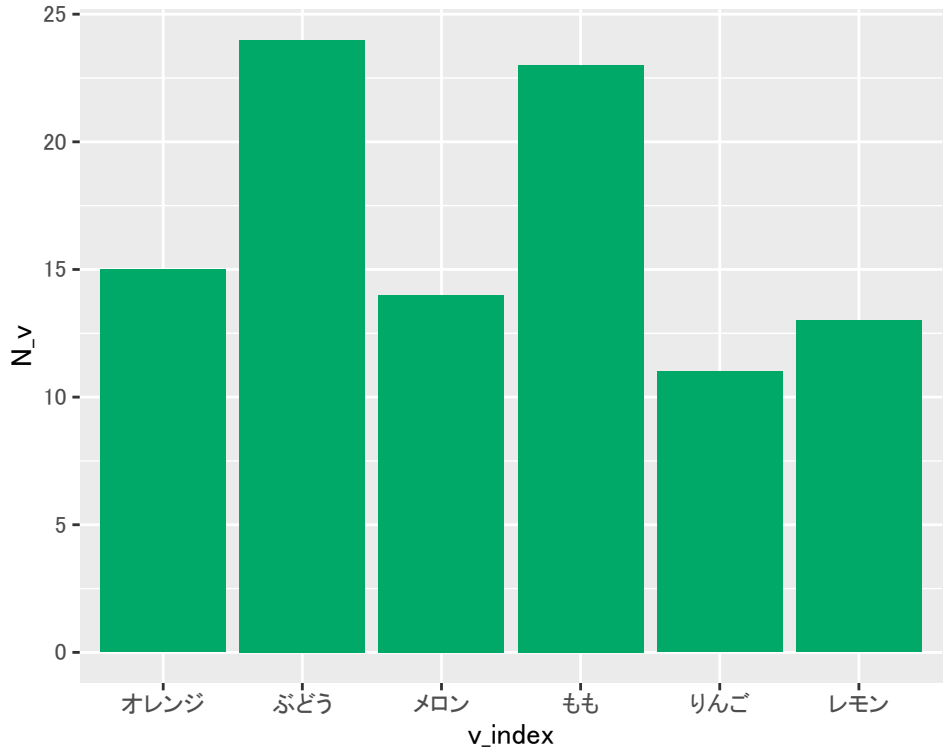
```
## w_n
## オレンジ　ぶどう　メロン　もも　りんご　レモン
##       15       24       14       23       11       13
```

```
# データフレームに変換する
N_v_df <- as_tibble(N_v_table)
colnames(N_v_df) <- c("v_index", "N_v")
N_v_df
```

```
## # A tibble: 6 x 2
##   v_index    N_v
##   <chr>    <int>
## 1 オレンジ    15
## 2 ぶどう      24
## 3 メロン      14
## 4 もも        23
## 5 りんご      11
## 6 レモン      13
```

生成された w_n(文書) に含まれる語彙を table() で集計します。
テーブル形式で返ってくるので、データフレームに変換します。

```
# 描画
ggplot(N_v_df, aes(v_index, N_v)) +
  geom_bar(stat = "identity", fill = "#00A968")
```



単語数の設定によって出現しない単語があるとその語彙は表に含まれません。

トピックモデルでは、このようにして文書 (各単語) が確率的に生成されている世界 (モデル) が想定されています。

Tips：単語と語彙の変換 (デルタ関数)

$$\prod_{n=1}^{N_d} \phi_{w_{d,n}} = \prod_{n=1}^{N_d} \prod_{v=1}^{V} \delta(w_{d,n} = v)\phi_v$$

で変換できます。
例文を使って確認しましょう。文書数は 1 なので、$d = 1$ です。

$$\begin{aligned}\prod_{v=1}^{6} \delta(w_{1,n} = v)\phi_v &= \Big(\delta(w_{1,1} = 1) * \phi_1\Big) * \Big(\delta(w_{1,1} = 2) * \phi_2\Big) * \cdots * \Big(\delta(w_{1,1} = 3) * \phi_6\Big) \\ &= (0 * \phi_1) * (0 * \phi_2) * (0 * \phi_3) * (1 * \phi_4) * (0 * \phi_5) * (0 * \phi_6) \\ &= \phi_4\end{aligned}$$

ではこれを全ての単語についてやってみましょう。

$$\begin{aligned}\prod_{n=1}^{9}\prod_{v=1}^{6} \delta(w_{1,n} = v)\phi_v &= \delta(w_{1,1} = 1)\phi_1 * \delta(w_{1,1} = 2)\phi_2 * \cdots * \delta(w_{1,1} = 3)\phi_6 \\ &\quad * \delta(w_{1,2} = 1)\phi_1 * \delta(w_{1,2} = 2)\phi_2 * \cdots * \delta(w_{1,2} = 3)\phi_6 \\ &\qquad \vdots \\ &\quad * \delta(w_{1,9} = 1)\phi_1 * \delta(w_{1,9} = 2)\phi_2 * \cdots * \delta(w_{1,9} = 3)\phi_6 \\ &= 0\phi_1 * 0\phi_2 * 0\phi_3 * 1\phi_4 * 0\phi_5 * 0\phi_6 \\ &\quad * 0\phi_1 * 0\phi_2 * 0\phi_3 * 0\phi_4 * 1\phi_5 * 0\phi_6 \\ &\qquad \vdots \\ &\quad * 0\phi_1 * 0\phi_2 * 1\phi_3 * 0\phi_4 * 0\phi_5 * 0\phi_6 \\ &= \phi_4 * \phi_5 * \phi_6 * \phi_2 * \phi_3 * \phi_1 * \phi_2 * \phi_4 * \phi_2 * \phi_3 \\ &= \phi_1 * \phi_2^3 * \phi_3^2 * \phi_4^2 * \phi_5 * \phi_6 \\ &= \prod_{v=1}^{6} \phi_v^{N_v}\end{aligned}$$

以上変換できました。

1.4 確率分布

1.4.1 カテゴリ分布

1.4.2 ディリクレ分布

2 ユニグラムモデル

例としてとある楽屋内の会話内容を扱いましたが、実際に分析する場合には複数の文書扱うことになります。それに合わせて表記を変えます。分析対象となる文書数を D として、文書 1 から D まで次元が増えます。単語番号や語彙番号と同様に、文書番号を示す添字を d として

$$d \in \{1, 2, \cdots, D\}$$

で表します。
文書集合を $\boldsymbol{W}$ として

$$\boldsymbol{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_D)$$

と表記することにします。また d 番目の文書 (単語集合) は $\boldsymbol{w}_d$ です。
更に、文書 d は N_d 個の単語の集合でしたので

$$\boldsymbol{w}_d = (w_{d,1}, w_{d,2}, \cdots, w_{d,N_d})$$

と表記します。$w_{d,n}$ は文書 d の n 番目の単語のことです。
文書ごとの単語数はバラバラです。表形式にまとめるために単語 (重複する形式) ではなく語彙 (重複しない形式) を使います。

	1	2	$\cdots$	V
1	$N_{1,1}$	$N_{1,2}$	$\cdots$	$N_{1,V}$
2	$N_{2,1}$	$N_{2,2}$	$\cdots$	$N_{2,V}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
D	$N_{D,1}$	$N_{D,2}$	$\cdots$	$N_{D,V}$

$N_{d,v}$ は文書 d において出現した語彙 v の数になります。

2.1 尤度

単語分布 $\boldsymbol{\phi}$ によって生成されるとする文書集合 $\boldsymbol{W}$ の生成確率は、

$$\begin{aligned} p(\boldsymbol{W}|\boldsymbol{\phi}) &= p(\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_D, |\boldsymbol{\phi}) \\ &= p(\boldsymbol{w}_1|\boldsymbol{\phi}) p(\boldsymbol{w}_2|\boldsymbol{\phi}) \cdots p(\boldsymbol{w}_D|\boldsymbol{\phi}) \\ &= \prod_{d=1}^{D} p(\boldsymbol{w}_d|\boldsymbol{\phi}) \end{aligned}$$

文書 1 から D までを掛け合わせたものになります。　では、文書 d の生成確率は

$$\begin{aligned} p(\boldsymbol{w}_d|\boldsymbol{\phi}) &= p(w_{d,1}, w_{d,2}, \cdots, w_{d,N_d}|\boldsymbol{\phi}) \\ &= p(w_{d,1}|\boldsymbol{\phi}) p(w_{d,2}|\boldsymbol{\phi}) \cdots p(w_{d,N_d}|\boldsymbol{\phi}) \\ &= \prod_{n=1}^{N_d} p(w_{d,n}|\boldsymbol{\phi}) \end{aligned}$$

その文書の全ての単語について掛け合わせたものになります。
文書 d の n 番目の単語の生成確率は

$$p(w_{d,n}|\boldsymbol{\phi}) = \phi_{w_{d,n}}$$

となります。
これを語彙で表現します。

$$\begin{aligned}\prod_{n=1}^{N_d} \phi_{w_{d,n}} &= \phi_{w_{d,1}}\phi_{w_{d,2}}\cdots\phi_{w_{d,N_d}} \\ &= \phi_1^{N_{d,1}}\phi_2^{N_{d,2}}\cdots\phi_V^{N_{d,V}} \\ &= \prod_{v=1}^{V}\phi_v^{N_{d,v}}\end{aligned}$$

重複しない単語 $w_{d,n}$ を語彙 v に置き換えると $N_{d,v}$ 回掛ければいいのでした。
従って、単語分布 $\boldsymbol{\phi}$ が所与の下での文書集合の生成確率 $p(\boldsymbol{W}|\boldsymbol{\phi})$ は

$$p(\boldsymbol{W}|\boldsymbol{\phi}) = \prod_{d=1}^{D}\prod_{v=1}^{V}\phi_v^{N_{d,v}}$$

と書き換えることができます。

2.1.1 記号一覧

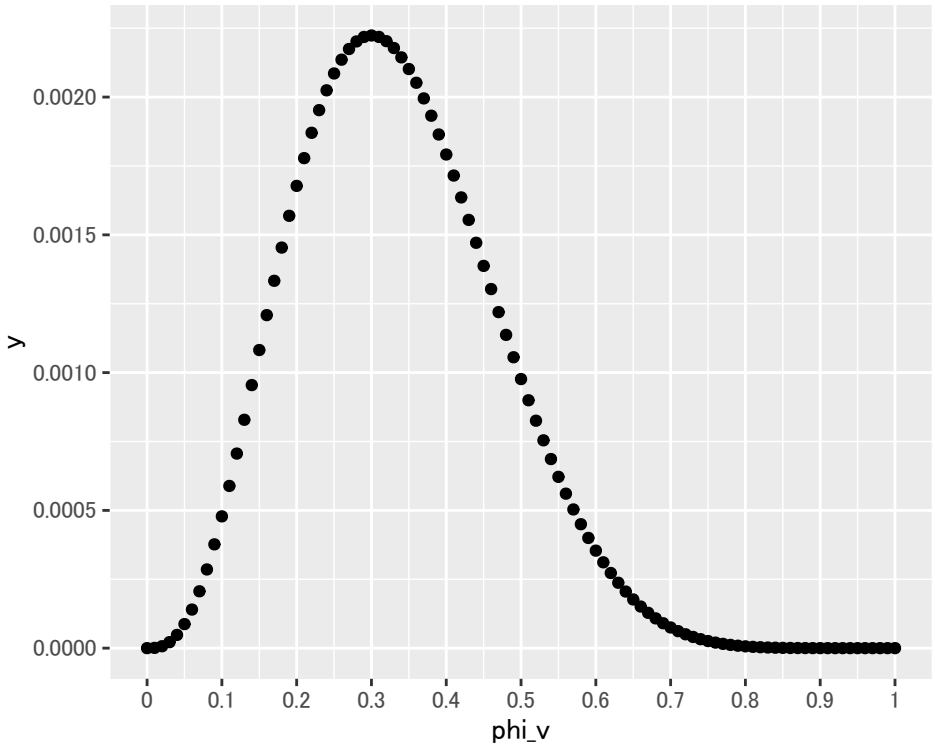
記号	意味	関係性
D	文書数	
$d \in \{1, 2, \cdots, D\}$	文書番号	
N	全文書での単語数	$N = \sum_{d=1}^{D} N_d = \sum_{v=1}^{V} N_v$
$n \in \{1, 2, \cdots, N_d\}$	単語番号	
N_d	文書 d の単語数	$N_d = \sum_{v=1}^{V} N_{d,v}$
N_v	全文書での語彙 v の出現回数	$N_v = \sum_{d=1}^{D} N_{d,v}$
$N_{d,v}$	文書 d での語彙 v の出現回数	
V	全文書での語彙数	
$v \in \{1, 2, \cdots, V\}$	語彙番号	
$\boldsymbol{W} = (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_d, \cdots, \boldsymbol{w}_D)$	文書集合	
$\boldsymbol{w}_d = (w_{d,1}, \cdots, w_{d,n}, \cdots, w_{d,N_d})$	文書 d の単語集合	
$w_{d,n} \in \{1, 2, \cdots, V\}$	文書 d の n 番目の単語	$w_{d,n} \sim \text{Categorical}(\boldsymbol{\phi})$
$\boldsymbol{\phi} = (\phi_1, \cdots, \phi_v, \cdots, \phi_V)$	単語分布	$\boldsymbol{\phi} \sim \text{Dirichlet}(\beta)$
ϕ_v	語彙 v の出現確率	$1 \geq \phi \geq 0,\ \sum_{v=1}^{V} \phi_v = 1$
β	単語分布のパラメータ	

2.2 最尤推定

サイコロの出目はカテゴリ分布に従います。

```
# 各語彙の出現回数
N_v <- c(1, 3, 2, 2, 1, 1)

```

```r
# 単語数
N <- sum(N_v)

# 語彙数を設定する
v <- 2

# 尤度を計算する
df <- tibble(
  phi_v = seq(0, 1, by = 0.01),
  phi_o = 1 - phi_v,
  y = phi_v^N_v[v] * phi_o^(N - N_v[v])
)

# 作図
ggplot(df, aes(phi_v, y)) +
  geom_point(fill = "#00A968") +
  scale_x_continuous(breaks = seq(0, 1, by = 0.1),
                     labels = seq(0, 1, by = 0.1))
```



2.3 MAP 推定

2.4 ベイズ更新

3　混合ユニグラムモデル

最初の例はミックスジュースの紹介文でしたので、果物に関連する単語しか出てきませんでした。他の文書でも果物について書かれているでしょうか。文書ごとに違うトピック (テーマ) を持つことが想定できます。

ユニグラムモデルでは、扱う文書全て 1 つのトピックに従って生成されている世界を想定していました。混合ユニグラムモデルで想定する世界では、複数のトピックの中から 1 つのトピックをそれぞれの文書を持ちます。各文書はそのトピックが持つ単語分布に従って単語が生成されているとします。

3.1　生成モデル

最初の例で用いたトピックとそのトピックが持つ語彙をトピック「果物」とします。その他に「色」「花」「名前」の 3 つのトピックとそのトピックが持つ語彙を 6 語ずつ設定しておきます。

color	flower	fruits	name
オレンジ	あやめ	オレンジ	あやか
もも	こぶし	ぶどう	かりん
ラベンダー	さくら	メロン	さくら
レモン	つばき	もも	ほのか
青緑	もも	りんご	まなか
白	ラベンダー	レモン	もも

それぞれ文書 (単語集合) ではなく、トピックとその単語をリスト化しただけであることに注意しましょう。この語彙それぞれの出現しやすさに応じて出現確率があり、確率変数として扱います。

また、トピック自体にも、テーマとして扱われやすさのように確率的に生成されます。

3.1.1　トピックの生成確率

トピックの数が K のとき、各トピックの確率を θ を用いて示すことにします[4]。

$$\boldsymbol{\theta} = (\theta_1, \theta_2, \cdots, \theta_K)$$

文書にトピック k が割り当てられる確率 θ_k は 0 から 1 の値をとり、トピック 1 になる確率からトピック K になる確率までを全て足し合わせると 1 になります。

$$0 \leq \theta_k \leq 1, \ \sum_{k=1}^{K} \theta_k = 1$$

k	トピック	$\boldsymbol{\theta}$	確率
1	色	θ_1	0.1
2	花	θ_2	0.3
3	果物	θ_3	0.4
4	名前	θ_4	0.2

[4] θ はパラメータでよく使われる記号ですが、ここではトピック (Topic) の t に相当するギリシャ文字 θ です。

それぞれこのように設定しておきます。

生成時には、各文書はトピックを持ち、それに従って文書が作られていると想定しています。しかし、各文書に割り当てられたトピックは、実際には観測できない情報です。これを潜在変数と呼びます。ここでは文書 d に割り当てられたトピックを z_d で表すことにします。全ての文書の潜在変数をまとめてトピック集合と呼び $\boldsymbol{z}$ と表記することにします。

$$\boldsymbol{z} = (z_1, z_2, \cdots, z_D)$$

トピック分布に従って、

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{\theta}) &= p(z_1, z_2, \cdots, z_D|\boldsymbol{\theta}) \\ &= p(z_1 = k|\boldsymbol{\theta})p(z_2 = k|\boldsymbol{\theta}) \cdots p(z_D = k|\boldsymbol{\theta}) \\ &= \prod_{d=1}^{D} p(z_d = k|\boldsymbol{\theta}) \end{aligned}$$

$$z_d \sim \mathrm{Cat}(\boldsymbol{\theta})$$

文書 d にトピック k が割り当てられる確率とは、つまりトピック k の確率のことなので θ_k のことです。

$$p(z_d = k|\boldsymbol{\theta}) = \theta_k$$

従って、トピック集合の生成確率は

$$p(\boldsymbol{z}|\boldsymbol{\theta}) = \prod_{d=1}^{D} \theta_k$$

となります。

3.1.2 文書集合の生成確率

ユニグラムモデルでは、トピックは 1 つだけでした。混合ユニグラムモデルでは、複数のトピックがあります。トピック (テーマ) ごとに単語の出現しやすさが変わるので、それぞれ別の単語分布になります。トピック数を K として

$$\boldsymbol{\Phi} = (\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_K)$$

になります。
単語分布ごとに 1 から V までの語彙の生成確率で構成されています。

$$\boldsymbol{\phi}_k = (\phi_{k,1}, \phi_{k,2}, \cdots, \phi_{k,V})$$

ユニグラムモデルと同様に、各語彙の生成確率は 0 から 1 の値をとり、総和は 1 になります。

$$0 \leq \phi_{k,v} \leq 1, \ \sum_{v=1}^{V} \phi_{k,v} = 1$$

これらをまとめると、文書集合 $\boldsymbol{W}$ は、割り当てられたトピック $\boldsymbol{z}$ 対応する単語分布 $\boldsymbol{\Phi}$ によって生成されているので

$$\begin{aligned} p(\boldsymbol{W}|\boldsymbol{z},\boldsymbol{\Phi}) &= p(\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_D|\boldsymbol{z},\boldsymbol{\Phi}) \\ &= p(\boldsymbol{w}_1|z_1=k,\boldsymbol{\phi}_k)p(\boldsymbol{w}_2|z_2=k,\boldsymbol{\phi}_k)\cdots p(\boldsymbol{w}_D|z_D=k,\boldsymbol{\phi}_k) \\ &= \prod_{d=1}^{D} p(\boldsymbol{w}_d|z_d=k,\boldsymbol{\phi}_k) \end{aligned}$$

文書 d にトピック k が与えられているので、トピック k が持つ単語分布 $\boldsymbol{\phi}_k$ を条件として、文書 $\boldsymbol{w}_d$ が生成されていることを表しています。

ユニグラムモデルと同様に、文書 (単語集合) は各単語の同時確率なので

$$\begin{aligned} p(\boldsymbol{w}_d|z_d=k,\boldsymbol{\phi}_k) &= p(w_{d,1},w_{d,2},\cdots,w_{d,N_d}|z_d=k,\boldsymbol{\phi}_k) \\ &= p(w_{d,1}|z_d=k,\boldsymbol{\phi}_k)p(w_{d,2}|z_d=k,\boldsymbol{\phi}_k)\cdots p(w_{d,N_d}|z_d=k,\boldsymbol{\phi}_k) \\ &= \prod_{n=1}^{N_d} p(w_{d,n}|z_d=k,\boldsymbol{\phi}_k) \end{aligned}$$

単語ごとの生成確率に分解できます。

各単語の生成確率は ϕ でした。トピック k において単語 $w_{d,n}$ の出現確率はそれらを添字として使い $\phi_{k,w_{d,n}}$ で表します。

$$p(w_{d,n}|z_d=k,\boldsymbol{\phi}_k) = \phi_{k,w_{d,n}}$$

重複を許す単語単位から重複しない語彙単位に書き換えます。

$$\begin{aligned} \prod_{n=1}^{N_d} p(w_{d,n}|z_d=k,\boldsymbol{\phi}_k) &= \phi_{k,w_{d,1}}\phi_{k,w_{d,2}}\cdots\phi_{k,w_{d,N_d}} \\ &= \phi_{k,1}^{N_{d,1}}\phi_{k,2}^{N_{d,2}}\cdots\phi_{k,V}^{N_{d,V}} \\ &= \prod_{v=1}^{V}\phi_{k,v}^{N_{d,v}} \end{aligned}$$

各語彙を出現回数 (重複している語数) 掛け合わせることで、変換できるのでました。

従って、単語分布 $\boldsymbol{\Phi}$ が所与の下での文書集合を計算可能な形に置き換えると

$$p(\boldsymbol{W}|\boldsymbol{z},\boldsymbol{\Phi}) = \prod_{d=1}^{D}\prod_{v=1}^{V}\phi_{k,v}^{N_{d,v}}$$

になります。

3.1.3 同時確率

トピック分布に従ってトピック集合が生成され、割り当てられたトピックに従って文書 (単語集合) が生成される混合ユニグラムモデルを同時確率で表現すると次のようになります。

$$p(\boldsymbol{W},\boldsymbol{z}|\boldsymbol{\Phi},\boldsymbol{\theta}) = p(\boldsymbol{W}|\boldsymbol{z},\boldsymbol{\Phi})p(\boldsymbol{z}|\boldsymbol{\theta})$$

更にこれを計算可能な形式で表現すると

$$
\begin{aligned}
p(\boldsymbol{W}, \boldsymbol{z}|\boldsymbol{\Phi}, \boldsymbol{\theta}) &= p(\boldsymbol{W}|\boldsymbol{z}, \boldsymbol{\Phi}) p(\boldsymbol{z}|\boldsymbol{\theta}) \\
&= \prod_{d=1}^{D} \sum_{k=1}^{K} \theta_k \prod_{v=1}^{V} \phi_{k,v}^{N_{d,v}}
\end{aligned}
$$

になります。

3.1.4 R でやってよう

```r
theta_k <- c(0.1, 0.3, 0.4, 0.2)
```

トピックとその語彙のベクトルを設定します。

次に、それぞれの語彙の出現確率を設定します。このプログラムでは出現確率を比率で指定しても大丈夫です。ここでは総和が 1 となるように設定しています。

```r
## 各トピックの単語を指定
# トピック 1：色
topic_color <- c(
  "オレンジ", "もも", "ラベンダー", "レモン", "青緑", "白"
)

# トピック 2：花
topic_flower <- c(
  "あやめ", "こぶし", "さくら", "つばき", "もも", "ラベンダー"
)

# トピック 3：果物
topic_fruits <- c(
  "オレンジ", "ぶどう", "メロン", "もも", "りんご", "レモン"
)

# トピック 4：人の名前
topic_name <- c(
  "あやか", "かりん", "さくら", "ほのか", "まなか", "もも"
)

v_index <- c(topic_color, topic_flower, topic_fruits, topic_name) %>% unique()
```

```r
## 各トピックの語彙と出現確率を設定する
# トピック 1：色
topic_color <- tibble(
  v_index = c("オレンジ", "もも", "ラベンダー", "レモン", "青緑", "白"),
  phi_1 = c(0.16667, 0.16667, 0.16667, 0.16667, 0.16667, 0.16667)
)

# トピック 2：花
topic_flower <- tibble(
  v_index = c("あやめ", "こぶし", "さくら", "つばき", "もも", "ラベンダー"),
  phi_2 = c(0.05, 0.1, 0.15, 0.2, 0.25, 0.25)
)
```

```r

# トピック 3：果物
topic_fruits <- tibble(
  v_index = c("オレンジ", "ぶどう", "メロン", "もも", "りんご", "レモン"),
  phi_3 = c(0.1, 0.15, 0.2, 0.3, 0.15, 0.1)
)

# トピック 4：人の名前
topic_name <- tibble(
  v_index = c("あやか", "かりん", "さくら", "ほのか", "まなか", "もも"),
  phi_4 = c(0.25, 0.25, 0.15, 0.15, 0.1, 0.1)
)
```

```r
Phi_df <- tibble(
  v_index = v_index
)
Phi_df
```

```
## # A tibble: 17 x 1
##    v_index
##    <chr>
##  1 オレンジ
##  2 もも
##  3 ラベンダー
##  4 レモン
##  5 青緑
##  6 白
##  7 あやめ
##  8 こぶし
##  9 さくら
## 10 つばき
## 11 ぶどう
## 12 メロン
## 13 りんご
## 14 あやか
## 15 かりん
## 16 ほのか
## 17 まなか
```

```r
Phi_df2 <- Phi_df %>%
  left_join(topic_color, key = "v_index") %>%
  left_join(topic_flower, key = "v_index") %>%
  left_join(topic_fruits, key = "v_index") %>%
  left_join(topic_name, key = "v_index")
Phi_df2[is.na((Phi_df2))] <- 0
Phi_df2
```

```
## # A tibble: 17 x 5
##    v_index    phi_1 phi_2 phi_3 phi_4
##    <chr>      <dbl> <dbl> <dbl> <dbl>
##  1 オレンジ   0.167  0     0.1   0
##  2 もも       0.167  0.25  0.3   0.1
##  3 ラベンダー 0.167  0.25  0     0
##  4 レモン     0.167  0     0.1   0
##  5 青緑       0.167  0     0     0
```

```
##  6 白          0.167  0     0     0
##  7 あやめ      0      0.05  0     0
##  8 こぶし      0      0.1   0     0
##  9 さくら      0      0.15  0     0.15
## 10 つばき      0      0.2   0     0
## 11 ぶどう      0      0     0.15  0
## 12 メロン      0      0     0.2   0
## 13 りんご      0      0     0.15  0
## 14 あやか      0      0     0     0.25
## 15 かりん      0      0     0     0.25
## 16 ほのか      0      0     0     0.15
## 17 まなか      0      0     0     0.1
```

```
# 単語分布
phi_kv <- Phi_df2[, -1] %>% as.matrix() %>% t()
rownames(phi_kv) <- NULL

# 結果の確認
phi_kv
```

```
##         [,1]    [,2]    [,3]    [,4]    [,5]    [,6] [,7] [,8] [,9] [,10]
## [1,] 0.16667 0.16667 0.16667 0.16667 0.16667 0.16667 0.00  0.0 0.00   0.0
## [2,] 0.00000 0.25000 0.25000 0.00000 0.00000 0.00000 0.05  0.1 0.15   0.2
## [3,] 0.10000 0.30000 0.00000 0.10000 0.00000 0.00000 0.00  0.0 0.00   0.0
## [4,] 0.00000 0.10000 0.00000 0.00000 0.00000 0.00000 0.00  0.0 0.15   0.0
##      [,11] [,12] [,13] [,14] [,15] [,16] [,17]
## [1,]  0.00   0.0  0.00  0.00  0.00  0.00   0.0
## [2,]  0.00   0.0  0.00  0.00  0.00  0.00   0.0
## [3,]  0.15   0.2  0.15  0.00  0.00  0.00   0.0
## [4,]  0.00   0.0  0.00  0.25  0.25  0.15   0.1
```

```
# トピックを生成する
z_d <- sample(1:4, size = 10, replace = TRUE, prob = theta_k)

# 結果の確認
z_d
```

```
##  [1] 4 2 4 4 3 2 2 3 3 3
```

```
# 語彙数
V <- length(v_index)


w_dn <- sample(1:V, size = 100, replace = TRUE, prob = phi_kv[1, ])
N_dv_table <- v_index[w_dn] %>% table()

# 結果の確認
N_dv_table
```

```
## .
##   オレンジ       もも ラベンダー      レモン       青緑         白
##         17         15         18         14         19         17
```

という要領で、10 個分の文書を生成しましょう。

```r
# 文書数を指定する
D <- 10

# 単語数を指定する
N_d <- 100

## 受け皿を用意しておく
# トピック集合
z_d <- rep(0, D)

# 文書集合
N_dv <- matrix(0, nrow = D, ncol = V,
               dimnames = list(paste0("d=", 1:D), v_index))

# 文書集合を生成する
for(d in 1:D) {

  # トピックを生成
  tmp_z_d <- rmultinom(n = 1, size = 1, prob = theta_k)
  z_d[d] <- which(tmp_z_d == 1)

  # 単語を生成
  N_dv[d, ] <- rmultinom(n = 1, size = N_d, prob = phi_kv[z_d[d], ])

}

# 結果の確認
z_d
```

```
##  [1] 3 2 4 1 3 3 4 4 2 3
```

```r
N_dv
```

```
##      オレンジ もも ラベンダー レモン 青緑 白 あやめ こぶし さくら つばき
## d=1        12   36          0     10    0  0      0      0      0      0
## d=2         0   19         30      0    0  0      4      9     22     16
## d=3         0    5          0      0    0  0      0      0     15      0
## d=4        14   20         22     14   12 18      0      0      0      0
## d=5        11   28          0     13    0  0      0      0      0      0
## d=6         5   26          0      8    0  0      0      0      0      0
## d=7         0    6          0      0    0  0      0      0     22      0
## d=8         0    6          0      0    0  0      0      0     21      0
## d=9         0   29         27      0    0  0      7     12     10     15
## d=10        8   34          0     15    0  0      0      0      0      0
##      ぶどう メロン りんご あやか かりん ほのか まなか
## d=1      11     19     12      0      0      0      0
## d=2       0      0      0      0      0      0      0
## d=3       0      0      0     25     32     13     10
## d=4       0      0      0      0      0      0      0
## d=5      10     26     12      0      0      0      0
## d=6      20     26     15      0      0      0      0
## d=7       0      0      0     21     30     14      7
## d=8       0      0      0     23     32      7     11
## d=9       0      0      0      0      0      0      0
```

```
## d=10     12     17     14      0      0      0      0
```

4　トピックモデル

Tips：トピック数の期待値 (デルタ関数)

$$N_{d,k} = \sum_{n=1}^{N_d} \delta(z_{d,n} = k)$$

$$\begin{aligned}\sum_{n=1}^{N_1} \delta(z_{1,n} = k) &= \delta(z_{1,1} = 1) + \delta(z_{1,2} = 1) + \cdots + \delta(z_{1,N_1} = 1) \\ &= 1 + 1 + 0 + 0 \\ &= 2 = N_{1,1}\end{aligned}$$

$$N_{k,v} = \sum_{d=1}^{D} \sum_{n=1}^{N_d} \delta(w_{d,n} = v, z_{d,n} = k)$$

$v = 1,\ k = 1$ について、つまり 1 番目の語彙の内、潜在トピック 1 が割り当てられた数を集計します。

$$\begin{aligned}\sum_{d=1}^{D} \sum_{n=1}^{N_d} \delta(w_{d,n} = 1, z_{d,n} = 1) &= \delta(w_{1,1} = 1, z_{1,1} = 1) + \delta(w_{1,2} = 1, z_{1,2} = 1) + \cdots + \delta(w_{1,N_1} = 1, z_{1,N_1} = 1) \\ &\quad + \delta(w_{2,1} = 1, z_{2,1} = 1) + \delta(w_{2,2} = 1, z_{2,2} = 1) + \cdots + \delta(w_{2,N_2} = 1, z_{2,N_2} = 1) \\ &\qquad\qquad \vdots \\ &\quad + \delta(w_{D,1} = 1, z_{D,1} = 1) + \delta(w_{D,2} = 1, z_{D,2} = 1) + \cdots + \delta(w_{D,N_D} = 1, z_{D,N_D} = 1) \\ &= 1 + 0 + 0 + 0 \\ &\quad + 1 + 0 + 0 + 0 \\ &\qquad\qquad \vdots \\ &\quad + 1 + 0 + 0 + 0 \\ &= 4 = N_{1,1}\end{aligned}$$

付録

4.1　基本的な計算例

4.1.1　対数

4.1.2　微分

おわりに

楽しかった！…よね？

Topic Models with Me ——ペンギンでも分かるトピックモデル——

2020 年 1 月 28 日　零版　第 0 刷
2020 年 0 月 00 日　初版　第 1 刷

著者 @anemptyarchive

発行者 anarchive-beta.com

製本 RStudio
