

## Topic Modeling - Latent Dirichlet Allocation

Filipe Rodrigues

Francisco Pereira

- Introduction
- In a nutshell
  - Example with traffic incidents
- Topic Modeling
- Concluding remarks

# Introduction

- Sometimes some variables of our  $x$  are in form of human language
- Often called **unstructured** data
- How to bring it into statistical models?

# Introduction

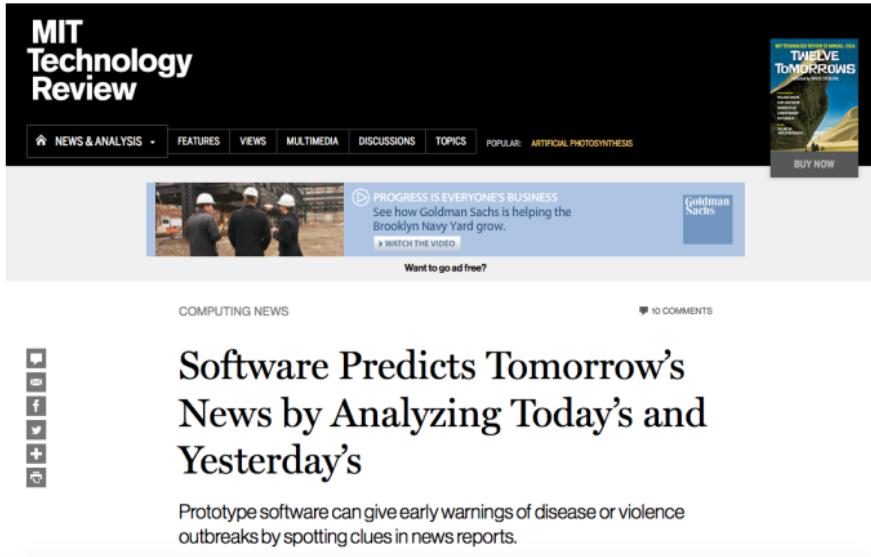
- Our goal is to have

$$\hat{y} = f(\mathbf{x}_{text}, \mathbf{x}_L)$$

where  $\mathbf{x}_{text}$  are variables coming from text data, and  $\mathbf{x}_L$  are non-textual variables

- we need to make  $\mathbf{x}_{text}$  usable in any multivariate model discussed in this course

# Motivation: News



The screenshot shows the homepage of MIT Technology Review. At the top, there is a navigation bar with links for NEWS & ANALYSIS, FEATURES, VIEWS, MULTIMEDIA, DISCUSSIONS, TOPICS, and POPULAR: ARTIFICIAL PHOTOSYNTHESIS. To the right of the navigation bar is a book cover for "MIT TECHNOLOGY REVIEW 21 MARCH 2014 TWELVE TOMORROWS" with a "BUY NOW" button. Below the navigation bar is a banner featuring two men in hard hats looking at a screen, with the text "PROGRESS IS EVERYONE'S BUSINESS See how Goldman Sachs is helping the Brooklyn Navy Yard grow." and a "WATCH THE VIDEO" link. A "Goldman Sachs" logo is also present. Below the banner, there is a "Want to go ad free?" link. The main content area has a "COMPUTING NEWS" section with a large headline: "Software Predicts Tomorrow's News by Analyzing Today's and Yesterday's". Below the headline is a subtext: "Prototype software can give early warnings of disease or violence outbreaks by spotting clues in news reports." On the left side of the main content area, there is a vertical column of social media sharing icons.

## MIT Technology Review

NEWS & ANALYSIS • FEATURES • VIEWS • MULTIMEDIA • DISCUSSIONS • TOPICS • POPULAR: ARTIFICIAL PHOTOSYNTHESIS

BUY NOW

PROGRESS IS EVERYONE'S BUSINESS

See how Goldman Sachs is helping the Brooklyn Navy Yard grow.

WATCH THE VIDEO

Goldman Sachs

Want to go ad free?

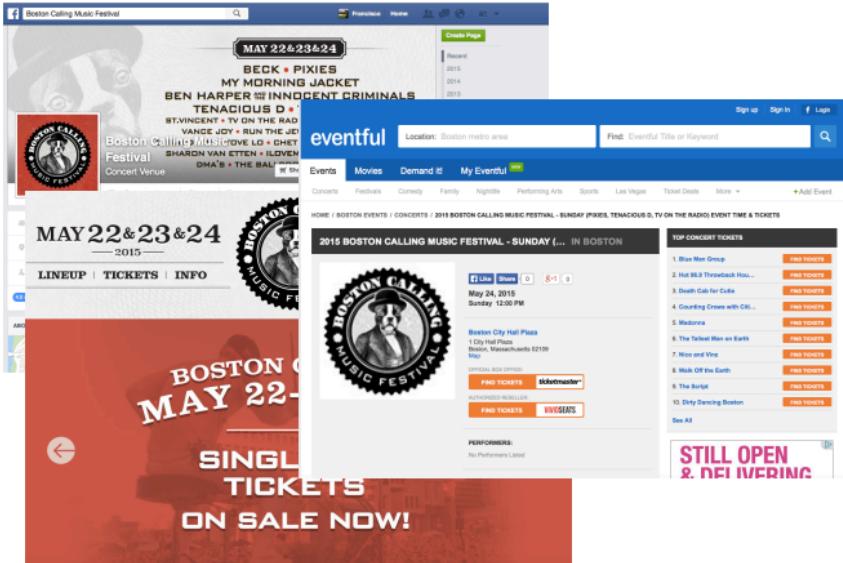
COMPUTING NEWS

10 COMMENTS

## Software Predicts Tomorrow's News by Analyzing Today's and Yesterday's

Prototype software can give early warnings of disease or violence outbreaks by spotting clues in news reports.

# Motivation: Special events

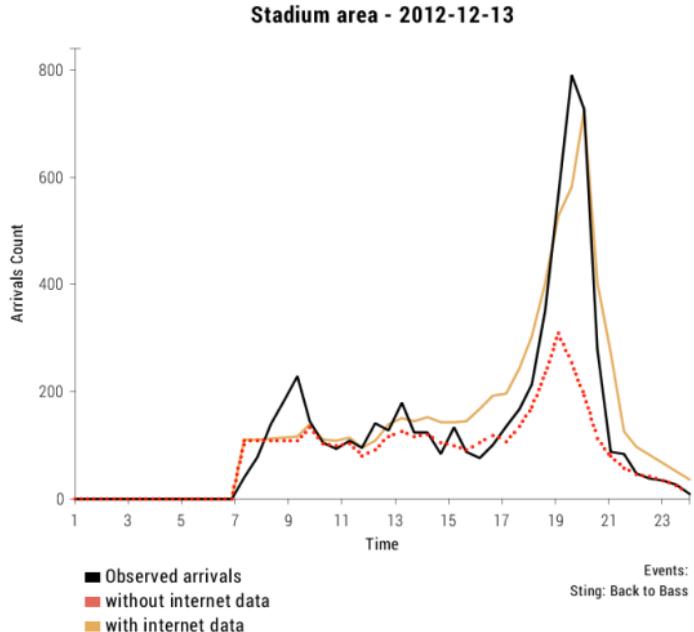


The image shows a composite screenshot of two web pages related to the Boston Calling Music Festival.

**Boston Calling Music Festival Website:** The left portion of the screenshot displays the official website for the festival. It features a banner for the "2015 BOSTON CALLING MUSIC FESTIVAL - SUNDAY (PIXIES, TENACIOUS D, TV ON THE RADIO) EVENT TIME & TICKETS". Below this, there's a large red section with the text "MAY 22-23-24 2015" and "SINGL TICKETS ON SALE NOW!". The site also includes sections for "LINEUP", "TICKETS", and "INFO".

**Eventful Search Results:** The right portion of the screenshot shows the results from the event search site Eventful. The search query "Boston Calling Music Festival" is entered. The results list "2015 BOSTON CALLING MUSIC FESTIVAL - SUNDAY (PIXIES, TENACIOUS D, TV ON THE RADIO) IN BOSTON" as the top result. This listing includes details like the date (May 24, 2015), time (Sunday 12:00 PM), location (Boston City Hall Plaza, 1 City Hall Plaza, Boston, Massachusetts 02110), and ticketing options through "OFFICIAL BOX OFFICE", "FAN TICKETS", and "VIVASEATS". To the right of the main listing, there's a sidebar titled "TOP CONCERT TICKETS" featuring links to other concert events.

# Motivation: Special events



# Motivation: traffic incidents



# Motivation: traffic incidents

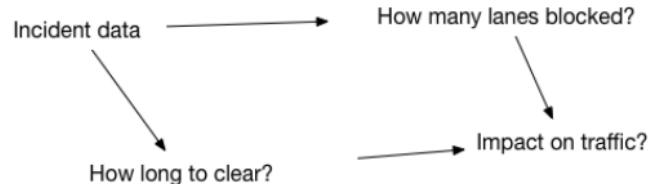
How many lanes blocked?

Impact on traffic?

How long to clear?



# Motivation: traffic incidents



# Motivation: traffic incidents



# Motivation: traffic incidents



## Incident report

Location type: 3  
Lane blockage: Lane 1, Shoulder blocked  
Down point: 20.32  
Congestion status: 0  
Queue length: 500m  
Start time: 2010-08-20 22:50:01      **Communications sequence**  
End time: 2010-08-20 23:31:45  
Number of vehicles: 2  
2250hrs - TP Joe X spots an accident. car and bike involved.  
2255hrs - Passers-by shift the bike to the shoulder.  
2300hrs - Ambulance arrives at location. LTM arrives at location.  
2309hrs - Ambulance conveys rider to National University Hospital.  
2310hrs - TP arrives at location.  
2311hrs - Notify by LTM the rider is seriously injured. The accident involves a car and bike.  
2331hrs - TP requests RC and LTM to resume patrolling. All other vehicles move off. Shoulder clear.

## In a nutshell: an example

- **Context:** Singapore Expressways, 2010-2012



- **Objective:** Predict duration of each incident
- **Data:** Logs of all incidents, 2010-2012

## In a nutshell: an example

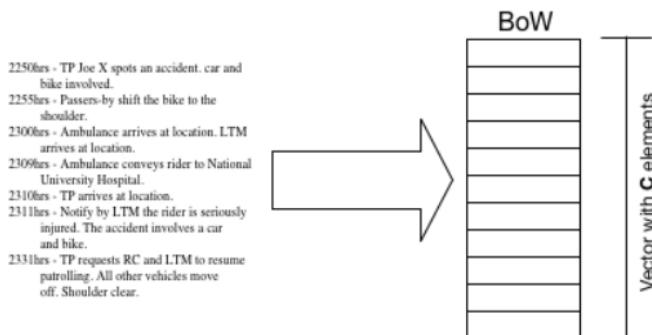
$$y = f(\mathbf{x}_{text}, \mathbf{x}_L)$$

with

$$\mathbf{x}_L = \begin{bmatrix} \mathbf{x}_{expressway\_ID} \\ \mathbf{x}_{num\_block\_lanes} \\ \mathbf{x}_{time\_of\_day} \\ \mathbf{x}_{day\_of\_week} \\ \mathbf{x}_{type\_of\_day} \\ \mathbf{x}_{congestion\_status} \\ \mathbf{x}_{num\_vehicles} \end{bmatrix}$$
$$\mathbf{x}_{text} = \begin{bmatrix} ? \\ ? \\ ? \end{bmatrix}$$

## In a nutshell: an example

- A document has a set of words, from a dictionary with size C
- How to represent it?
  - The bag-of-words (BoW) representation
  - Dimension C
  - Vector of word frequencies



## BoW Representation

- 2250hrs - TP Joe X spots an accident. car and bike involved.
- 2255hrs - Passers-by shift the bike to the shoulder.
- 2300hrs - Ambulance arrives at location. LTM arrives at location.
- 2309hrs - Ambulance conveys rider to National University Hospital.
- 2310hrs - TP arrives at location.
- 2311hrs - Notify by LTM the rider is seriously injured. The accident involves a car and bike.
- 2331hrs - TP requests RC and LTM to resume patrolling. All other vehicles move off. Shoulder clear.

BoW

# BoW Representation

Document

2250hrs - **TP Joe X** spots an **accident**. **car** and **bike** involved.  
2255hrs - Passers-by shift the **bike** to the **shoulder**.  
2300hrs - Ambulance arrives at **location**. **LTM** arrives at **location**.  
2309hrs - Ambulance conveys **rider** to National University Hospital.  
2310hrs - **TP** arrives at **location**.  
2311hrs - Notify by **LTM** the **rider** is **seriously injured**. The **accident** involves a **car** and **bike**.  
2331hrs - **TP** requests **RC** and **LTM** to resume patrolling. All other vehicles move off. Shoulder clear.

3	TP
1	Joe X
1	spots
2	accident
2	car
3	bike
1	passers-by
1	shoulder
2	ambulance
3	LTM
3	location
2	rider
1	seriously
1	injured
...	...
...	...
...	...
...	...

## BoW representation - data preparation

- Replace words by their *stems* (e.g. injury, injured, injuries → injur)
- Remove words from the **stopwords** list (e.g. *and, or, that, it...*)
- The dictionary is a collection of C words (i.e. their stems)
- Calculate frequency of all words (ignore word sequence)

## From text to regression

- We could use BoW frequencies directly in a model:

$$y = f(\mathbf{x}_{bow}, \mathbf{x}_L)$$

where  $\mathbf{x}_{bow}$  is the BoW vector

- Dimensionality= $C+L$ , can be huge ( $C \simeq 10^6$  for English)!
- Many words for same concept (e.g. synonyms, acronyms, abbreviations)
- Many words related (e.g. "ambulance", "injury", "hospital")
- Many words irrelevant (e.g. proper names)
- Manual work (e.g. replace synonyms with same word)?
  - VERY time consuming and prone to error!

# Topic modeling

- Each document describes a set of  $K$  concepts (or **topics**),  $K \ll C$
- Each topic is *latent*, i.e. we cannot observe it

## In a nutshell: an example

### Some examples of topics from our incident reports dataset

```
topic #1 - 0.085*tp + 0.071*convei + 0.070*ab + 0.064*tow + 0.062*hosp + 0.056*rider + 0.048*bike
topic #2 - 0.054*ir + 0.049*confirm + 0.045*congest + 0.034*traffic + 0.027*case + 0.023*soe + 0.018*heavi
topic #3 - 0.001*veh + 0.001*nch + 0.001*spot + 0.000*rc + 0.000*clear + 0.000*itm + 0.000*1ln
topic #6 - 0.138*tow + 0.122*noinjur + 0.092*veh + 0.080*present + 0.066*clear + 0.061*owner +
0.047*nodamag
topic #11 - 0.101*assist + 0.056*taxi + 0.045*notifi + 0.032*requir + 0.029*came + 0.029*p3 + 0.027*passeng
topic #12 - 0.076*polic + 0.041*otm + 0.033*2nd + 0.033*last + 0.023*div + 0.022*tm1 + 0.016*1st
topic #13 - 0.077*damag + 0.071*call + 0.043*skid + 0.034*tp + 0.029*near + 0.028*self + 0.027*vig
topic #14 - 0.072*left + 0.057*activ + 0.050*tp + 0.045*2ln + 0.040*scdf + 0.035*open + 0.029*spill
topic #15 - 0.160*minor + 0.049*exchang + 0.047*particular + 0.032*advis + 0.029*refus + 0.025*detail +
0.013*involv2
topic #18 - 0.112*report + 0.093*rc + 0.063*tm + 0.041*btw + 0.036*lorri + 0.034*bef + 0.032*actv
```

# Topic modeling

- Each document describes a set of  $K$  concepts (or **topics**),  $K \ll C$
- Each topic is *latent*, i.e. we cannot observe it
- Examples of topic “labels”:
  - Medical assistance
  - Significant damage
  - Low damage
  - Oil spillage
  - ...

# In a nutshell: an example

## *Medical assistance*

```
topic #1 - 0.085*tp + 0.071*convei + 0.070*ab + 0.064*tow + 0.062*hosp + 0.056*rider + 0.048*bike
topic #2 - 0.054*ir + 0.049*confirm + 0.045*congest + 0.034*traffic + 0.027*case + 0.023*soe + 0.018*heavi
topic #3 - 0.001*veh + 0.001*nch + 0.001*spot + 0.000*rc + 0.000*clear + 0.000*itm + 0.000*1ln
topic #6 - 0.138*tow + 0.122*noinjur + 0.092*veh + 0.080*present + 0.066*clear + 0.061*owner +
0.047*nodamag
topic #11 - 0.101*assist + 0.056*taxi + 0.045*notifi + 0.032*requir + 0.029*came + 0.029*p3 + 0.027*passeng
topic #12 - 0.076*polic + 0.041*otm + 0.033*2nd + 0.033*last + 0.023*div + 0.022*tm1 + 0.016*1st
topic #13 - 0.077*damag + 0.071*call + 0.042*skid + 0.034*tp + 0.029*near + 0.028*self + 0.027*vig
topic #14 - 0.072*left + 0.057*activ + 0.050*tp + 0.045*2ln + 0.040*scdf + 0.035*open + 0.029*spill
topic #15 - 0.160*minor + 0.049*exchang + 0.047*particular + 0.032*advis + 0.029*refus + 0.025*detail +
0.013*involv2
topic #18 - 0.112*report + 0.093*rc + 0.063*tm + 0.041*btw + 0.036*lorri + 0.034*bef + 0.032*actv
```

# In a nutshell: an example

## *Significant damage*

```
topic #1 - 0.085*tp + 0.071*convei + 0.070*ab + 0.064*tow + 0.062*hosp + 0.056*rider + 0.048*bike
topic #2 - 0.054*ir + 0.049*confirm + 0.045*congest + 0.034*traffic + 0.027*case + 0.023*soe + 0.018*heavi
topic #3 - 0.001*veh + 0.001*nch + 0.001*spot + 0.000*rc + 0.000*clear + 0.000*ltm + 0.000*1ln
topic #6 - 0.138*tow + 0.122*noinjur + 0.092*veh + 0.080*present + 0.066*clear + 0.061*owner +
0.047*nodamag
topic #11 - 0.101*assist + 0.056*taxi + 0.045*notifi + 0.032*requir + 0.029*came + 0.029*p3 + 0.027*passeng
topic #12 - 0.076*polic + 0.041*otm + 0.033*2nd + 0.033*last + 0.023*div + 0.022*tm1 + 0.016*1st
topic #13 - 0.077*damag + 0.071*call + 0.043*skid + 0.034*tp + 0.029*near + 0.028*self + 0.027*vig
topic #14 - 0.072*left + 0.057*activ + 0.050*tp + 0.045*2ln + 0.040*scdf + 0.035*open + 0.029*spill
topic #15 - 0.160*minor + 0.049*exchang + 0.047*particular + 0.032*advis + 0.029*refus + 0.025*detail +
0.013*involv2
topic #18 - 0.112*report + 0.093*rc + 0.063*tm + 0.041*btw + 0.036*lorri + 0.034*bef + 0.032*actv
```

# In a nutshell: an example

## Low damage

```
topic #1 - 0.085*tp + 0.071*convei + 0.070*ab + 0.064*tow + 0.062*hosp + 0.056*rider + 0.048*bike
topic #2 - 0.054*ir + 0.049*confirm + 0.045*congest + 0.034*traffic + 0.027*case + 0.023*soe + 0.018*heavi
topic #3 - 0.001*veh + 0.001*nch + 0.001*spot + 0.000*rc + 0.000*clear + 0.000*ltm + 0.000*1ln
topic #6 - 0.138*tow + 0.122*noinjur + 0.092*veh + 0.080*present + 0.066*clear + 0.061*owner +
0.047*nodamag
topic #11 - 0.101*assist + 0.056*taxi + 0.045*notifi + 0.032*requir + 0.029*came + 0.029*p3 + 0.027*passeng
topic #12 - 0.076*polic + 0.041*otm + 0.033*2nd + 0.033*last + 0.023*div + 0.022*tm1 + 0.016*1st
topic #13 - 0.077*damag + 0.071*call + 0.043*skid + 0.034*tp + 0.029*near + 0.028*self + 0.027*vig
topic #14 - 0.072*left + 0.057*activ + 0.050*tp + 0.045*2ln + 0.040*scdf + 0.035*open + 0.029*spill
topic #15 - 0.160*minor + 0.049*exchang + 0.047*particular + 0.032*advis + 0.029*refus + 0.025*detail +
0.013*involv2
topic #18 - 0.112*report + 0.093*rc + 0.063*tm + 0.041*btw + 0.036*lorri + 0.034*bef + 0.032*actv
```

# In a nutshell: an example

## *Oil spillage*

```
topic #1 - 0.085*tp + 0.071*convei + 0.070*ab + 0.064*tow + 0.062*hosp + 0.056*rider + 0.048*bike
topic #2 - 0.054*ir + 0.049*confirm + 0.045*congest + 0.034*traffic + 0.027*case + 0.023*soe + 0.018*heavi
topic #3 - 0.001*veh + 0.001*nch + 0.001*spot + 0.000*rc + 0.000*clear + 0.000*itm + 0.000*1ln
topic #6 - 0.138*tow + 0.122*noinjur + 0.092*veh + 0.080*present + 0.066*clear + 0.061*owner +
0.047*nodamag
topic #11 - 0.101*assist + 0.056*taxi + 0.045*notifi + 0.032*requir + 0.029*came + 0.029*p3 + 0.027*passeng
topic #12 - 0.076*polic + 0.041*otm + 0.033*2nd + 0.033*last + 0.023*div + 0.022*tm1 + 0.016*1st
topic #13 - 0.077*damag + 0.071*call + 0.043*skid + 0.034*tp + 0.029*near + 0.028*self + 0.027*vig
topic #14 - 0.072*left + 0.057*activ + 0.050*tp + 0.045*2ln + 0.040*scdf + 0.035*open + 0.029*spill
topic #15 - 0.160*minor + 0.049*exchang + 0.047*particular + 0.032*advis + 0.029*refus + 0.025*detail +
0.013*involv2
topic #18 - 0.112*report + 0.093*rc + 0.063*tm + 0.041*btw + 0.036*lorri + 0.034*bef + 0.032*actv
```

## In a nutshell: an example

### Some vehicle types (*taxis, truck/lorry*)

```
topic #1 - 0.085*tp + 0.071*convei + 0.070*ab + 0.064*tow + 0.062*hosp + 0.056*rider + 0.048*bike
topic #2 - 0.054*ir + 0.049*confirm + 0.045*congest + 0.034*traffic + 0.027*case + 0.023*soe + 0.018*heavi
topic #3 - 0.001*veh + 0.001*nch + 0.001*spot + 0.000*rc + 0.000*clear + 0.000*ltm + 0.000*1ln
topic #6 - 0.138*tow + 0.122*noinjur + 0.092*veh + 0.080*present + 0.066*clear + 0.061*owner +
0.047*nodamag
topic #11 - 0.101*assist + 0.056*taxi + 0.045*notifi + 0.032*requir + 0.029*came + 0.029*p3 + 0.027*passeng
topic #12 - 0.076*polic + 0.041*otm + 0.033*2nd + 0.033*last + 0.023*div + 0.022*tm1 + 0.016*1st
topic #13 - 0.077*damag + 0.071*call + 0.043*skid + 0.034*tp + 0.029*near + 0.028*self + 0.027*vig
topic #14 - 0.072*left + 0.057*activ + 0.050*tp + 0.045*2ln + 0.040*scdf + 0.035*open + 0.029*spill
topic #15 - 0.160*minor + 0.049*exchang + 0.047*particular + 0.032*advis + 0.029*refus + 0.025*detail +
0.013*involv2
topic #18 - 0.112*report + 0.093*rc + 0.063*tm + 0.041*btw + 0.036*lrorri + 0.034*bef + 0.032*actv
```

# Topic modeling

- Can we quantify **how much** of each topic exists in a document?
  - E.g. 20% medical assistance, 50% significant damage, 30% oil spillage
  - A document would be a **linear combination** of  $K$  topics!
  - We go from  $C$  to  $K \rightarrow$  **dimensionality reduction**

$$\mathbf{x}_{text} = \begin{bmatrix} \mathbf{x}_{topic\_1} \\ \mathbf{x}_{topic\_2} \\ \dots \\ \mathbf{x}_{topic\_K} \end{bmatrix}$$

# Topic modeling

- Each topic:
  - Itself a vector with dimensionality  $C$  (i.e. one entry per word in dictionary)
  - Inferred from an unsupervised process (called *Latent Dirichlet Allocation*, LDA)
  - One can see a topic as a *prototypical document*

## In a nutshell: an example

### Topic assignment in our example text

2250hrs - TP Joe X spots an accident. car and bike involved.  
2255hrs - Passers-by shift the bike to the shoulder.  
2300hrs - Ambulance arrives at location. LTM arrives at location.  
2309hrs - Ambulance conveys rider to National University Hospital.  
2310hrs - TP arrives at location.  
2311hrs - Notify by LTM the rider is seriously injured. The accident involves a car and bike.  
2331hrs - TP requests RC and LTM to resume patrolling. All other vehicles move off. Shoulder clear.



0.43	topic #1
0.07	topic #3
...	...
0.10	topic #12
0.12	topic #18

# Latent Dirichlet Allocation (LDA)

Generative model without probability distributions yet

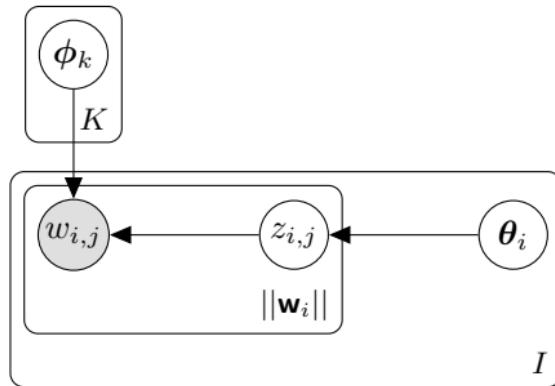
We want to generate a set of  $I$  documents ( $i = 1, 2, \dots, I$ ), each one being a sequence of words,  $\mathbf{w}_i$ :

- ① For each topic  $k$  ( $k=1\dots K$ ), there is a vector  $\phi_k$  of  $C$  words (dictionary)
- ② For each document  $i$ :
  - a) We assign it a vector  $\theta_i$  of  $K$  topic proportions
  - b) For each of the words,  $j$  ( $j = 1, 2, \dots, ||\mathbf{w}_i||$ ), we have
    - i) A topic assignment,  $z_{i,j}$  ( $z_{i,j} \in \{1, 2, \dots, K\}$ ) according to  $\theta_i$
    - ii) A word,  $w_{i,j}$ , according to  $\phi_{z_{i,j}}$

# Latent Dirichlet Allocation (LDA)

Generative model without probability distributions yet

- ① For each topic  $k$  ( $k=1\dots K$ ), there is a vector  $\phi_k$  of  $C$  words (dictionary)
- ② For each document  $i$ :
  - a) We assign it a vector  $\theta_i$  of  $K$  topic proportions
  - b) For each of the words,  $j$  ( $j = 1, 2, \dots, ||\mathbf{w}_i||$ ), we have
    - i) A topic assignment,  $z_{i,j}$  ( $z_{i,j} \in \{1, 2, \dots, K\}$ ) according to  $\theta_i$
    - ii) A word,  $w_{i,j}$ , according to  $\phi_{z_{i,j}}$



## Some background

- Categorical distribution,  $\text{Cat}(\theta)$

- With A categories

$$p(x = a|\theta) = \theta_a$$

where  $\theta = [\theta_1, \theta_2, \dots, \theta_A]$

$$\sum_{a=1}^A \theta_a = 1$$

- When A=2, it's called *bernoulli* distribution
- Equivalent to *multinomial* with 1 trial

## Some background

- Dirichlet distribution
  - Random vector,  $\theta$ , of dimension A;

$$\theta \sim \text{Dir}(\alpha)$$

- $\alpha_1, \alpha_2, \dots, \alpha_A$ , with  $\alpha_i > 0$

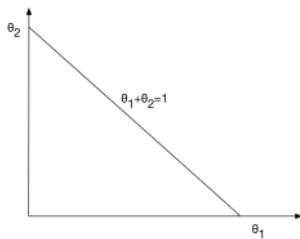
$$p(\theta|\alpha) = \text{Dir}(\alpha) = \frac{\Gamma\left(\sum_{i=1}^A \alpha_i\right)}{\prod_{i=1}^A \Gamma(\alpha_i)} \prod_{i=1}^A \theta_i^{\alpha_i - 1}$$

where  $\sum_{i=1}^A \theta_i = 1$  and  $\theta_i \in [0, 1]$ .

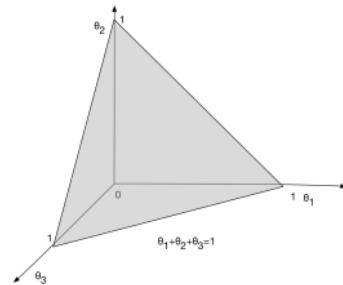
- $\Gamma(u)$  is the gamma function, a generalization of the factorial function, such that  $\Gamma(u + 1) = u\Gamma(u)$

## Some background - Dirichlet distribution

- It can be viewed as a probability distribution on an  $A - 1$  dimensional simplex



1-D Simplex (segment)

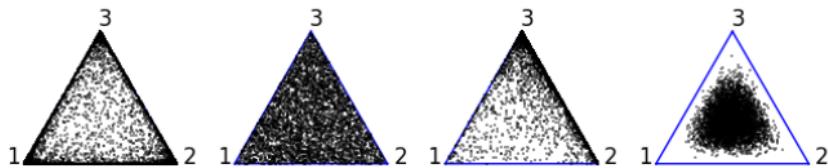


2-D Simplex (triangle)

# Topic modeling

- A sample from the Dirichlet distribution with dimension  $A$  is a vector  $\theta$ , such that  $\sum_{i=1}^A \theta_i = 1$

$$\alpha = (0.200, 0.200, 0.200) \quad \alpha = (1.000, 1.000, 1.000) \quad \alpha = (0.100, 0.500, 0.990) \quad \alpha = (5.000, 5.000, 5.000)$$



- Perfect to generate arguments for the categorical distribution!

# Playtime!



- Open "08. LDA.ipynb" notebook
- Do part 1
- Estimated time: 15 minutes

# Latent Dirichlet Allocation (LDA)

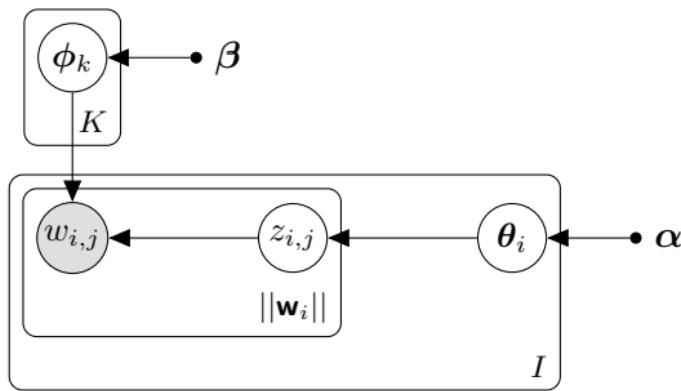
- ① For each topic  $k$ , there is a vector of  $C$  words,  $\phi_k$ , such that  $p(\phi_k|\beta) = \text{Dir}(\beta)$

# Latent Dirichlet Allocation (LDA)

- ① For each topic  $k$ , there is a vector of  $C$  words,  $\phi_k$ , such that  $p(\phi_k|\beta) = \text{Dir}(\beta)$
- ② For each document  $i$ :
  - a) There is a vector of  $K$  topics,  $\theta_i$ , such that  $p(\theta_i|\alpha) = \text{Dir}(\alpha)$
  - b) For each of the words,  $j$ , in each document  $i$  ( $j = 1, 2, \dots, ||\mathbf{w}_i||$ ), we have
    - i) A topic assignment,  $z_{i,j}$  ( $z_{i,j} \in \{1, 2, \dots, K\}$ ) such that  $p(z_{i,j}|\theta_i) = \text{Cat}(\theta_i)$
    - ii) A word,  $w_{i,j}$ , such that  $p(w_{i,j}|\phi_{z_{i,j}}) = \text{Cat}(\phi_{z_{i,j}})$

# Latent Dirichlet Allocation (LDA)

- ① For each topic  $k$ , there is a vector of  $C$  words,  $\phi_k$ , such that  $p(\phi_k|\beta) = \text{Dir}(\beta)$
- ② For each document  $i$ :
  - a) There is a vector of  $K$  topics,  $\theta_i$ , such that  $p(\theta_i|\alpha) = \text{Dir}(\alpha)$
  - b) For each of the words,  $j$ , in each document  $i$  ( $j = 1, 2, \dots, ||\mathbf{w}_i||$ ), we have
    - i) A topic assignment,  $z_{i,j}$  ( $z_{i,j} \in \{1, 2, \dots, K\}$ ) such that  $p(z_{i,j}|\theta_i) = \text{Cat}(\theta_i)$
    - ii) A word,  $w_{i,j}$ , such that  $p(w_{i,j}|\phi_{z_{i,j}}) = \text{Cat}(\phi_{z_{i,j}})$



# Playtime!



- Open "08. LDA.ipynb" notebook
- Do part 2
- Estimated time: 30 minutes

# Topic modeling - LDA

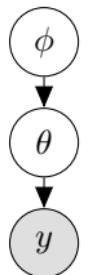
Joint distribution:

$$p(\mathbf{W}, \mathbf{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi} | \boldsymbol{\alpha}, \boldsymbol{\beta}) = \left( \prod_{i=1}^I p(\boldsymbol{\theta}_i | \boldsymbol{\alpha}) \prod_{j=1}^{||\mathbf{w}_i||} p(w_{i,j} | \boldsymbol{\phi}_{z_{i,j}}) p(z_{i,j} | \boldsymbol{\theta}_i) \right) \prod_{k=1}^K p(\boldsymbol{\phi}_k | \boldsymbol{\beta})$$

- No analytical solution
- Common implementations with Gibbs sampling or Variational Bayes

# Discrete Latent Variables in STAN

- In its present version, one cannot explicitly use discrete latent variables in STAN
- Limitation with its MCMC inference process (explained in upcoming class)
- For example



$$\phi \sim \text{Beta}(2, 1)$$

$$\theta \sim \text{Bernoulli}(\phi)$$

$$y \sim \mathcal{N}(k\theta, 10)$$

$$p(y, \theta, \phi) = p(y|\theta) p(\theta|\phi) p(\phi)$$

# Discrete Latent Variables in STAN

- It would allow, however



$$p(y, \phi) = \sum_{\theta} p(y|\theta)p(\theta|\phi)p(\phi)$$

- Marginalization of the latent variable
- We "get rid" of the latent variable, but use its distribution correctly
- How do we do this in STAN?
- STAN takes care of the priors (because all their parameters are known!)
- So we need to focus on the likelihood

$$L(y) = p(y|\phi) = \sum_{\theta} p(y|\theta)p(\theta|\phi)$$

# Discrete Latent Variables in STAN

- Notice that:

$$\theta \sim \text{Bernoulli}(\phi) \implies p(\theta|\phi) = \phi^\theta (1-\phi)^{(1-\theta)}$$

- So we have:

$$\begin{aligned} L(y) &= p(y|\phi) = \sum_{\theta} p(y|\theta)p(\theta|\phi) = \sum_{\theta} \left( \mathcal{N}(y|k\theta, 10) \phi^\theta (1-\phi)^{(1-\theta)} \right) = \\ &= \phi \mathcal{N}(y|k, 10) + (1-\phi) \mathcal{N}(y|0, 10) \end{aligned}$$

- STAN uses **log-probabilities**, so we should have:

$$LL(y) = \log \left( \phi \mathcal{N}(y|k, 10) + (1-\phi) \mathcal{N}(y|0, 10) \right)$$

# Discrete Latent Variables in STAN

$$LL(y) = \log \left( \phi \mathcal{N}(y|k, 10) + (1 - \phi) \mathcal{N}(y|0, 10) \right)$$

- In fact, you'll need to use the function `log_sum_exp(vector v)`: *real*
  - Receives a vector **v** of real values
  - Transforms each element,  $v_i$ , of the vector with  $e^{v_i}$
  - Calculates the "log sum" to all elements  $v_i$
  - So, to use it, we need to apply  $\log(v_i)$  at each element
  - The model from above in STAN:

```
model = {
    phi ~ beta(2,1); // prior

    // likelihood
    real gamma[2];
    gamma[1] = log(phi) + normal_lpdf(y| k, 10);
    gamma[2] = log(1-phi) + normal_lpdf(y| 0, 10);
    target += log_sum_exp(gamma);
}
```

# Playtime!



- Open "08. LDA.ipynb" notebook
- Do part 3
- Estimated time: 1 hour

## Back to the nutshell: our example

$$y = f(\mathbf{x}_{text}, \mathbf{x}_L)$$

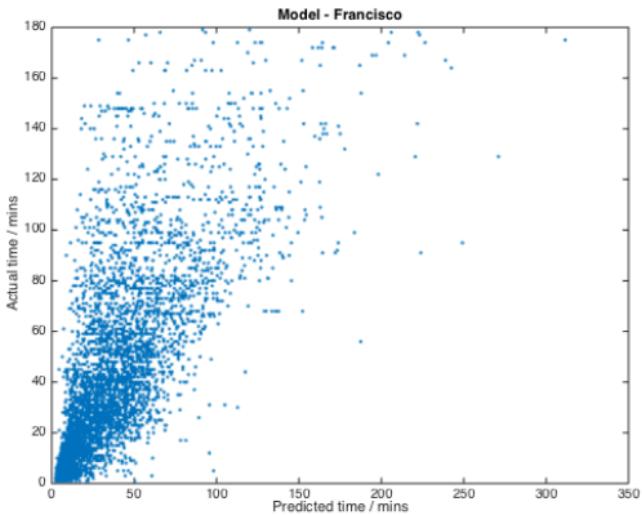
with

$$\mathbf{x}_L = \begin{bmatrix} \mathbf{x}_{expressway\_ID} \\ \mathbf{x}_{num\_block\_lanes} \\ \mathbf{x}_{time\_of\_day} \\ \mathbf{x}_{day\_of\_week} \\ \mathbf{x}_{type\_of\_day} \\ \mathbf{x}_{congestion\_status} \\ \mathbf{x}_{num\_vehicles} \end{bmatrix}$$
$$\mathbf{x}_{text} = \begin{bmatrix} \mathbf{x}_{topic\_1} \\ \mathbf{x}_{topic\_2} \\ \dots \\ \mathbf{x}_{topic\_K} \end{bmatrix}$$

# Incident Duration Prediction

- Data:
  - 10000+ incidents
  - Each report modeled using 25 topics
  - Original information on time of day, location etc.
- Goal:
  - regression model for incident duration
- An accurate prediction for the duration is important for modelling the impact on traffic

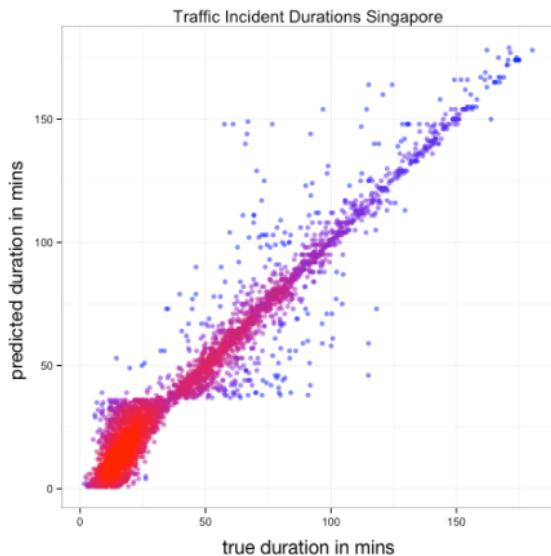
## Non-parametric model ( $R^2=0.461$ )



- Predicted vs. Actual durations for test data.
- Without textual data,  $R^2 = 0.230$

# Incident Duration Prediction ( $R^2=0.603$ )

A slightly more complicated (non-parametric) model...



## Concluding remarks

- In practice (e.g. sklearn), documents for LDA use BoW representation
  - Mechanics is exactly the same
  - We only mention so you know how to do it :-)
- LDA is super-optimized today - Variational approximation algorithm

# Concluding remarks

- Topic modeling falls in the general family of *mixed membership* models
  - Each document is a mixture of components
- Similarities with Principal Components Analysis (PCA), Factor Analysis...
- What you saw was **unsupervised**...
  - What would be the supervised version?

# Reference material

Introduction to Probabilistic Topic Models.

David Blei, 2011 (pp 1-8)

[www.cs.princeton.edu/~blei/papers/Blei2011.pdf](http://www.cs.princeton.edu/~blei/papers/Blei2011.pdf)

*More detail:*

Probabilistic Topic Models, Mark Steyvers and Tom Griffiths

In T. Landauer, D McNamara, S. Dennis, and W. Kintsch (eds), Latent Semantic Analysis: A Road to Meaning. Laurence Erlbaum

[psiexp.ss.uci.edu/research/papers/SteyversGriffithsLSABookFormatted.pdf](http://psiexp.ss.uci.edu/research/papers/SteyversGriffithsLSABookFormatted.pdf)

# Appendix



- Another example: Demand prediction for special events

# Demand prediction for special events

- **Context:**

- Singapore November 2011- February 2012
- 3 venues (Singapore Indoor Stadium, Kallang Theatre, Singapore Expo)

- **Objective:**

- Predict number of public transport arrivals (in 30 min blocks)
- Useful for operations planning (e.g. reallocate drivers)

- **Data:**

- EZLink (tap-in/tap-out) data, November 2011-February 2012
- 1500+ Event data from Eventful.org, Facebook, Google (600+ for selected venues)

# Demand prediction for special events

- Model variables
  - Time of day (discrete and continuous), day of week, type of day
  - Time to next event/time since event started (in 30 min slots)
  - Number of Google hits, Facebook likes, Wikipedia page (dummy)
  - Topics from textual description ( $K = 25$ )
- Regression model(s)
  - Linear regression
  - Kernel method (Gaussian Processes)

# Demand prediction for special events

