
Ida Documentation

Ida Developers

Sep 09, 2018

Contents

1	Getting started	3
2	Installing lida	7
2.1	Windows	7
2.2	Mac OS X	7
2.3	Linux	7
2.4	Installation from source	8
3	API	9
3.1	lida.lida	9
3.2	lida.utils	9
4	Contributing	11
5	Style Guidelines	13
6	Building in Develop Mode	15
7	Groups	17
8	What's New	19
8.1	v1.0.5 (18. June 2017)	19
8.2	v1.0.4 (13. July 2016)	19
8.3	v1.0.3 (5. Nov 2015)	19
9	Indices and tables	21

lda implements latent Dirichlet allocation (LDA) using collapsed Gibbs sampling. **lda** is fast and can be installed without a compiler on Linux, OS X, and Windows.

The interface follows conventions found in [scikit-learn](#). The following demonstrates how to inspect a model of a subset of the Reuters news dataset. (The input below, *X*, is a document-term matrix.)

```
>>> import numpy as np
>>> import lda
>>> X = lda.datasets.load_reuters()
>>> vocab = lda.datasets.load_reuters_vocab()
>>> titles = lda.datasets.load_reuters_titles()
>>> X.shape
(395, 4258)
>>> X.sum()
84010
>>> model = lda.LDA(n_topics=20, n_iter=1500, random_state=1)
>>> model.fit(X) # model.fit_transform(X) is also available
>>> topic_word = model.topic_word_ # model.components_ also works
>>> n_top_words = 8
>>> for i, topic_dist in enumerate(topic_word):
...     topic_words = np.array(vocab)[np.argsort(topic_dist)[: -n_top_words: -1]]
...     print('Topic {}: {}'.format(i, ' '.join(topic_words)))
Topic 0: british churchill sale million major letters west
Topic 1: church government political country state people party
Topic 2: elvis king fans presley life concert young
Topic 3: yeltsin russian russia president kremlin moscow michael
Topic 4: pope vatican paul john surgery hospital pontiff
Topic 5: family funeral police miami versace cunanan city
Topic 6: simpson former years court president wife south
Topic 7: order mother successor election nuns church nirmala
Topic 8: charles prince diana royal king queen parker
Topic 9: film french france against bardot paris poster
Topic 10: germany german war nazi letter christian book
Topic 11: east peace prize award timor quebec belo
Topic 12: n't life show told very love television
Topic 13: years year time last church world people
Topic 14: mother teresa heart calcutta charity nun hospital
Topic 15: city salonika capital buddhist cultural vietnam byzantine
Topic 16: music tour opera singer israel people film
Topic 17: church catholic bernardin cardinal bishop wright death
Topic 18: harriman clinton u.s ambassador paris president churchill
Topic 19: city museum art exhibition century million churches
```

Contents:

CHAPTER 1

Getting started

The following demonstrates how to inspect a model of a subset of the Reuters news dataset. The input below, *X*, is a document-term matrix (sparse matrices are accepted).

```
>>> import numpy as np
>>> import lda
>>> X = lda.datasets.load_reuters()
>>> vocab = lda.datasets.load_reuters_vocab()
>>> titles = lda.datasets.load_reuters_titles()
>>> X.shape
(395, 4258)
>>> X.sum()
84010
>>> model = lda.LDA(n_topics=20, n_iter=1500, random_state=1)
>>> model.fit(X) # model.fit_transform(X) is also available
>>> topic_word = model.topic_word_ # model.components_ also works
>>> n_top_words = 8
>>> for i, topic_dist in enumerate(topic_word):
...     topic_words = np.array(vocab)[np.argsort(topic_dist)[: -n_top_words: -1]]
...     print('Topic {}: {}'.format(i, ' '.join(topic_words)))
Topic 0: british churchill sale million major letters west
Topic 1: church government political country state people party
Topic 2: elvis king fans presley life concert young
Topic 3: yeltsin russian russia president kremlin moscow michael
Topic 4: pope vatican paul john surgery hospital pontiff
Topic 5: family funeral police miami versace cunanan city
Topic 6: simpson former years court president wife south
Topic 7: order mother successor election nuns church nirmala
Topic 8: charles prince diana royal king queen parker
Topic 9: film french france against bardot paris poster
Topic 10: germany german war nazi letter christian book
Topic 11: east peace prize award timor quebec belo
Topic 12: n't life show told very love television
Topic 13: years year time last church world people
Topic 14: mother teresa heart calcutta charity nun hospital
```

(continues on next page)

(continued from previous page)

```

Topic 15: city salonika capital buddhist cultural vietnam byzantine
Topic 16: music tour opera singer israel people film
Topic 17: church catholic bernardin cardinal bishop wright death
Topic 18: harriman clinton u.s ambassador paris president churchill
Topic 19: city museum art exhibition century million churches
    
```

The document-topic distributions are available in `model.doc_topic_`.

```

>>> doc_topic = model.doc_topic_
>>> for i in range(10):
...     print("{} (top topic: {})".format(titles[i], doc_topic[i].argmax()))
0 UK: Prince Charles spearheads British royal revolution. LONDON 1996-08-20 (top_
  ↳topic: 8)
1 GERMANY: Historic Dresden church rising from WW2 ashes. DRESDEN, Germany 1996-08-21_
  ↳(top topic: 13)
2 INDIA: Mother Teresa's condition said still unstable. CALCUTTA 1996-08-23 (top_
  ↳topic: 14)
3 UK: Palace warns British weekly over Charles pictures. LONDON 1996-08-25 (top_
  ↳topic: 8)
4 INDIA: Mother Teresa, slightly stronger, blesses nuns. CALCUTTA 1996-08-25 (top_
  ↳topic: 14)
5 INDIA: Mother Teresa's condition unchanged, thousands pray. CALCUTTA 1996-08-25_
  ↳(top topic: 14)
6 INDIA: Mother Teresa shows signs of strength, blesses nuns. CALCUTTA 1996-08-26_
  ↳(top topic: 14)
7 INDIA: Mother Teresa's condition improves, many pray. CALCUTTA, India 1996-08-25_
  ↳(top topic: 14)
8 INDIA: Mother Teresa improves, nuns pray for "miracle". CALCUTTA 1996-08-26 (top_
  ↳topic: 14)
9 UK: Charles under fire over prospect of Queen Camilla. LONDON 1996-08-26 (top_
  ↳topic: 8)
    
```

Document-topic distributions may be inferred for out-of-sample texts using the `transform` method:

```

>>> X = lda.datasets.load_reuters()
>>> titles = lda.datasets.load_reuters_titles()
>>> X_train = X[10:]
>>> X_test = X[:10]
>>> titles_test = titles[:10]
>>> model = lda.LDA(n_topics=20, n_iter=1500, random_state=1)
>>> model.fit(X_train)
>>> doc_topic_test = model.transform(X_test)
>>> for title, topics in zip(titles_test, doc_topic_test):
...     print("{} (top topic: {})".format(title, topics.argmax()))
0 UK: Prince Charles spearheads British royal revolution. LONDON 1996-08-20 (top_
  ↳topic: 7)
1 GERMANY: Historic Dresden church rising from WW2 ashes. DRESDEN, Germany 1996-08-21_
  ↳(top topic: 11)
2 INDIA: Mother Teresa's condition said still unstable. CALCUTTA 1996-08-23 (top_
  ↳topic: 4)
3 UK: Palace warns British weekly over Charles pictures. LONDON 1996-08-25 (top_
  ↳topic: 7)
4 INDIA: Mother Teresa, slightly stronger, blesses nuns. CALCUTTA 1996-08-25 (top_
  ↳topic: 4)
5 INDIA: Mother Teresa's condition unchanged, thousands pray. CALCUTTA 1996-08-25_
  ↳(top topic: 4)
6 INDIA: Mother Teresa shows signs of strength, blesses nuns. CALCUTTA 1996-08-26_
  ↳(top topic: 4)
    
```

(continues on next page)

(continued from previous page)

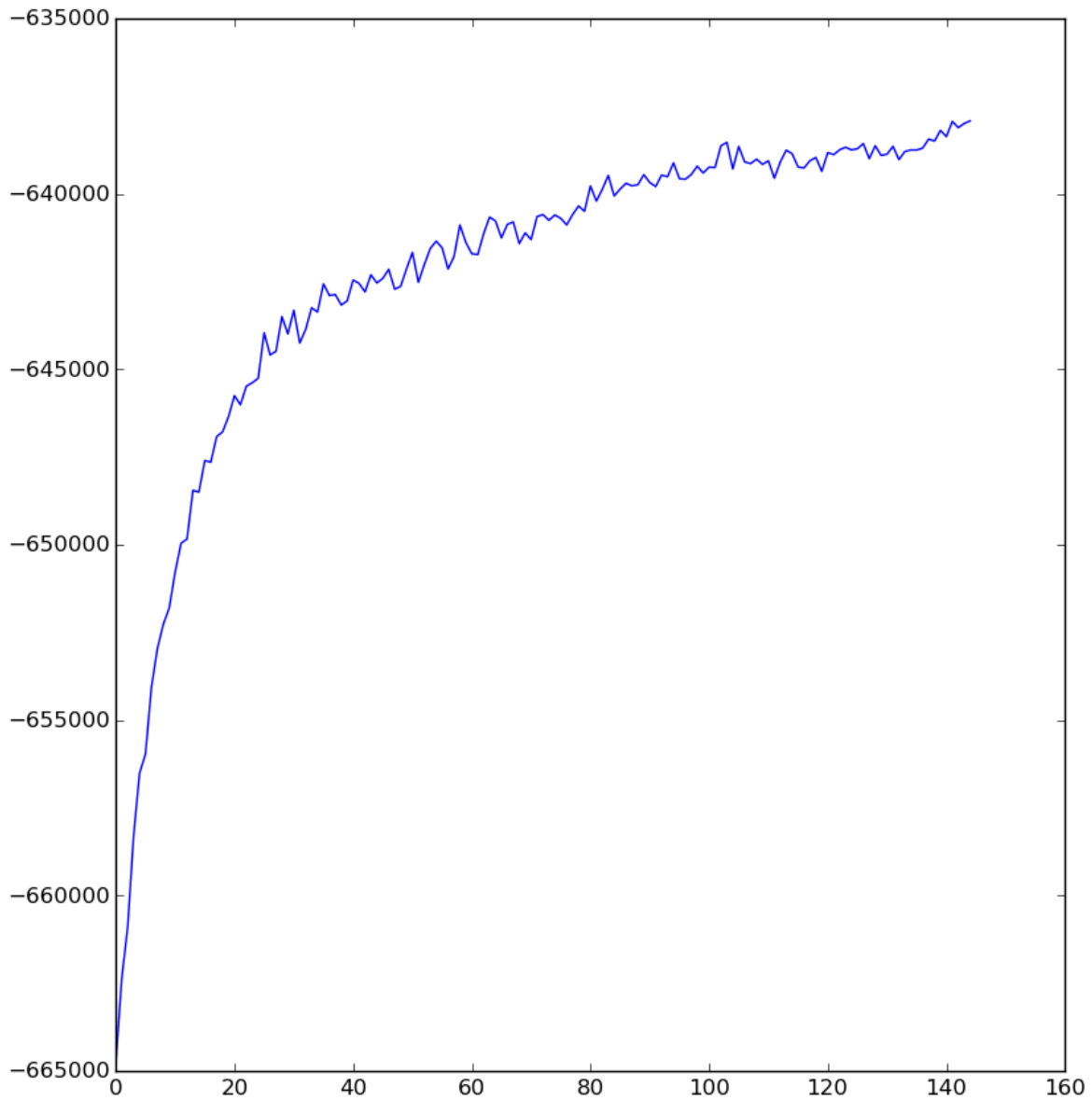
```
7 INDIA: Mother Teresa's condition improves, many pray. CALCUTTA, India 1996-08-25_
↳(top topic: 4)
8 INDIA: Mother Teresa improves, nuns pray for "miracle". CALCUTTA 1996-08-26 (top_
↳topic: 4)
9 UK: Charles under fire over prospect of Queen Camilla. LONDON 1996-08-26 (top_
↳topic: 11)
```

(Note that the topic numbers have changed due to LDA not being an **identifiable** model. The phenomenon is known as **label switching** in the literature.)

Convergence may be monitored by accessing the `loglikelihoods_` attribute on a fitted model. The attribute is bound to a list which records the sequence of log likelihoods associated with the model at different iterations (thinned by the `refresh` parameter).

(The following code assumes `matplotlib` is installed.)

```
>>> import matplotlib.pyplot as plt
>>> # skipping the first few entries makes the graph more readable
>>> plt.plot(model.loglikelihoods_[5:])
```



Judging convergence from the plot, the model should be fit with a slightly greater number of iterations.

lda requires Python (≥ 2.7 or ≥ 3.3) and NumPy ($\geq 1.6.1$). If these requirements are satisfied, lda should install successfully with:

```
pip install lda
```

If you encounter problems, consult the platform-specific instructions below.

2.1 Windows

lda and its dependencies are all available as wheel packages for Windows:

```
pip install lda
```

2.2 Mac OS X

lda and its dependencies are all available as wheel packages for Mac OS X:

```
pip install lda
```

2.3 Linux

lda and its dependencies are all available as wheel packages for most distributions of Linux:

```
pip install lda
```

2.4 Installation from source

Installing from source requires you to have installed the Python development headers and a working C/C++ compiler. Under Debian-based operating systems, which include Ubuntu, you can install all these requirements by issuing:

```
sudo apt-get install build-essential python3-dev python3-setuptools \  
python3-numpy
```

Before attempting a command such as `python setup.py install` you will need to run Cython to generate the relevant C files:

```
make cython
```

3.1 Ida.Ida

3.2 Ida.utils

CHAPTER 4

Contributing

CHAPTER 5

Style Guidelines

Before contributing a patch, please read the Python “Style Commandments” written by the OpenStack developers:
<http://docs.openstack.org/developer/hacking/>

CHAPTER 6

Building in Develop Mode

To build in develop mode on OS X, first install Cython and pbr. Then run:

```
git clone https://github.com/ariddell/lda.git
cd lda
make cython
python setup.py develop
```


CHAPTER 7

Groups

The `lda-users` group is for general discussion of `lda`, including modeling and installation issues:

- [lda users group](#)

You can subscribe by sending an e-mail to `lda-users+subscribe@googlegroups.com`.

8.1 v1.0.5 (18. June 2017)

- Wheels for Python 3.6

8.2 v1.0.4 (13. July 2016)

- Linux wheels (manylinux1)

8.3 v1.0.3 (5. Nov 2015)

- Python 3.5 wheels
- Release GIL during sampling
- Many minor fixes

CHAPTER 9

Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)