03_dirichlet_distribution

September 29, 2021

1 Dirichlet Distribution

1.1 Imports

[1]:
```
import warnings
from collections import OrderedDict
from pathlib import Path

import numpy as np
import pandas as pd

# Visualization
from ipywidgets import interact, FloatSlider
import matplotlib.pyplot as plt
import seaborn as sns
```

[3]:
```
%matplotlib inline
plt.style.use('ggplot')
plt.rcParams['figure.figsize'] = (14.0, 8.7)
warnings.filterwarnings('ignore')
pd.options.display.float_format = '{:,.2f}'.format
```

1.2 Simulate Dirichlet Distribution

The Dirichlet distribution produces probability vectors that can be used with discrete distributions. That is, it randomly generates a given number of values that are positive and sum to one. It has a parameter of positive real value that controls the concentration of the probabilities. Values closer to zero mean that only a few values will be positive and receive most probability mass.

The following simulation let's you interactively explore how different parameter values affect the resulting probability distributions.

[4]:
```
f=FloatSlider(value=1, min=1e-2, max=1e2, step=1e-2, continuous_update=False,␣
 ↪description='Alpha')
@interact(alpha=f)
def sample_dirichlet(alpha):
    topics = 10
    draws= 9
```

```
    alphas = np.full(shape=topics, fill_value=alpha)
    samples = np.random.dirichlet(alpha=alphas, size=draws)

    fig, axes = plt.subplots(nrows=3, ncols=3, sharex=True, sharey=True)
    axes = axes.flatten()
    plt.setp(axes, ylim=(0, 1))
    for i, sample in enumerate(samples):
        axes[i].bar(x=list(range(10)), height=sample, color=sns.
↪color_palette("Set2", 10))
    fig.suptitle('Dirichlet Allocation | 10 Topics, 9 Samples')
    fig.tight_layout()
    plt.subplots_adjust(top=.95)
```

```
interactive(children=(FloatSlider(value=1.0, continuous_update=False,␣
↪description='Alpha', min=0.01, step=0.01…
```

[ ]: