Modified Dirichlet Distribution: Allowing Negative Parameters to Induce Stronger Sparsity[*]

Kewei Tu
School of Information Science and Technology
ShanghaiTech University, Shanghai, China
tukw@shanghaitech.edu.cn

Abstract

The Dirichlet distribution (Dir) is one of the most widely used prior distributions in statistical approaches to natural language processing. The parameters of Dir are required to be positive, which significantly limits its strength as a sparsity prior. In this paper, we propose a simple modification to the Dirichlet distribution that allows the parameters to be negative. Our modified Dirichlet distribution (mDir) not only induces much stronger sparsity, but also simultaneously performs smoothing. mDir is still conjugate to the multinomial distribution, which simplifies posterior inference. We introduce two simple and efficient algorithms for finding the mode of mDir. Our experiments on learning Gaussian mixtures and unsupervised dependency parsing demonstrate the advantage of mDir over Dir.

1 Dirichlet Distribution

The Dirichlet distribution (Dir) is defined over probability vectors $\boldsymbol{x} = \langle x_1, \ldots, x_n \rangle$ with positive parameter vector $\boldsymbol{\alpha} = \langle \alpha_1, \ldots, \alpha_n \rangle$:

$$\mathrm{Dir}(\boldsymbol{x}; \boldsymbol{\alpha}) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{i=1}^{n} x_i^{\alpha_i - 1}$$

where the normalization factor $\mathrm{B}(\boldsymbol{\alpha})$ is the multivariate beta function. When the elements in $\boldsymbol{\alpha}$ are larger than one, Dir can be used as a smoothness prior that prefers more uniform probability vectors, with larger $\boldsymbol{\alpha}$ values inducing more smoothness. When the elements in $\boldsymbol{\alpha}$ are less than one, Dir can be seen as a sparsity prior that prefers sparse probability vectors, with smaller $\boldsymbol{\alpha}$ values inducing stronger sparsity. To better understand its sparsity preference, we take the logarithm of Dir:

$$\log \mathrm{Dir}(\boldsymbol{x}; \boldsymbol{\alpha}) = \sum_{i=1}^{n} (\alpha_i - 1) \log x_i + \mathrm{constant}$$

Since $\alpha_i - 1$ is negative, the closer x_i is to zero, the higher the log probability becomes. The coefficient $\alpha_i - 1$ controls the strength of the sparsity preference. However, α_i is required to be positive in Dir because otherwise the normalization factor becomes divergent. Consequently, the strength of the sparsity preference is upper bounded. This becomes problematic when a strong prior is needed, for instance, when the training dataset is large relative to the model size (e.g., in unsupervised learning of an unlexicalized probabilistic grammar) and thus the likelihood may dominate the posterior without a strong prior.

2 Modified Dirichlet Distribution

We make a simple modification to the Dirichlet distribution that allows the parameters in $\boldsymbol{\alpha}$ to become negative. To handle the divergent normalization factor, we require that each x_i must be lower bounded by a small positive constant ϵ. Our modified Dirichlet distribution (mDir) is defined as follows.

$$\mathrm{mDir}(\boldsymbol{x}; \boldsymbol{\alpha}, \epsilon) = \begin{cases} 0 & \text{if } \exists i, x_i < \epsilon \\ \frac{1}{Z(\boldsymbol{\alpha}, \epsilon)} \prod_{i=1}^{n} x_i^{\alpha_i - 1} & \text{otherwise} \end{cases}$$

where we require $0 < \epsilon \leq \frac{1}{n}$ and do *not* require α_i to be positive. With fixed values of $\boldsymbol{\alpha}$ and ϵ, the

[*] This work was supported by the National Natural Science Foundation of China (61503248).

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1986–1991,
Austin, Texas, November 1-5, 2016. ©2016 Association for Computational Linguistics

unnormalized probability density is always bounded and hence the normalization factor $Z(\boldsymbol{\alpha}, \epsilon)$ is finite.

It is easy to show that mDir is still conjugate to the multinomial distribution. Similar to Dir, mDir can be used as a smoothness/sparsity prior depending on the values of $\boldsymbol{\alpha}$. Because $\boldsymbol{\alpha}$ is no longer required to be positive, we can achieve very strong sparsity preference by using a highly negative vector of $\boldsymbol{\alpha}$. Note that here we no longer achieve sparsity in its strict sense; instead, by sparsity we mean most elements in $\boldsymbol{x}$ reach their lower bound ϵ. Parameter ϵ can thus be seen as a smoothing factor that prevents any element in $\boldsymbol{x}$ to become too small. Therefore, with proper parameters, mDir is able to simultaneously achieve sparsity and smoothness. This can be useful in many applications where one wants to learn a sparse multinomial distribution in an iterative way without premature pruning of components.

2.1 Finding the Mode

If $\forall i, \alpha_i - 1 \leq 0$, then the mode of mDir can be shown to be:

$$x_i = \begin{cases} 1-(n-1)\epsilon & \text{if } i = \arg\max_i \alpha_i \\ \epsilon & \text{otherwise} \end{cases}$$

Otherwise, we can find the mode with Algorithm 1. The algorithm first lets $x_i = \epsilon$ if $\alpha_i \leq 1$ and otherwise lets x_i be proportional to $\alpha_i - 1$. It then looks for variables in $\boldsymbol{x}$ that are less than ϵ, increases them to ϵ, and renormalizes the rest of the variables. The renormalization may decrease some additional variables below ϵ, so the procedure is repeated until all the variables are larger than or equal to ϵ.

Theorem 1. *If $\exists i, \alpha_i > 1$, then Algorithm 1 correctly finds a mode of* $\mathrm{mDir}(\boldsymbol{x}; \boldsymbol{\alpha}, \epsilon)$.

Proof. First, we can show that for any i such that $\alpha_i \leq 1$, we must have $x_i = \epsilon$ at the mode. This is because if $x_i > \epsilon$, then we can increase the probability density by first decreasing x_i to ϵ (hence increasing $x_i^{\alpha_i - 1}$), and then increasing some other variable x_j with $\alpha_j > 1$ to satisfy the normalization condition (hence also increasing $x_j^{\alpha_j - 1}$). This is consistent with the output of the algorithm.

Once we fix the value to ϵ for any variable x_i s.t. $\alpha_i \leq 1$, the log probability density function becomes strictly concave on the simplex specified by the linear constraints $\sum_i x_i = 1$ and $x_i \geq \epsilon$. The strict concavity can be proved by showing that the log probability density function is twice differentiable and the Hessian is negative definite at any point of the simplex.

Algorithm 1 Mode-finding of $\mathrm{mDir}(\boldsymbol{x}; \boldsymbol{\alpha}, \epsilon)$

```
S ← {i|α_i ≤ 1}
T ← ∅
repeat
    T ← T ⋃ S
    for i ∈ T do
        x_i ← ϵ
    end for
    z ← Σ_{i∉T}(α_i − 1)
    for i ∉ T do
        x_i ← (α_i−1)/z × (1 − ϵ|T|)
    end for
    S ← {i|x_i < ϵ}
until S = ∅
return ⟨x_1, ..., x_n⟩
```

With a concave function and linear constraints, the KKT conditions are sufficient for optimality. We need to show that the output of the algorithm satisfies the following KKT conditions:

- Stationarity: $\forall i, \frac{\alpha_i - 1}{x_i} = -\mu_i + \lambda$
- Primal feasibility: $\forall i, x_i \geq \epsilon$ and $\sum_i x_i = 1$
- Dual feasibility: $\forall i, \mu_i \geq 0$
- Complementary slackness: $\forall i, \mu_i(x_i - \epsilon) = 0$

Let $x_i^{(k)}$ and $T^{(k)}$ be the values of x_i and T after k iterations of the algorithm. Suppose the algorithm terminates after K iterations. So the output of the algorithm is $\langle x_1^{(K)}, \ldots, x_n^{(K)} \rangle$, which we will prove satisfies the KKT conditions.

For any i s.t. $x_i^{(K)} > \epsilon$, we set $\mu_i = 0$ and $\lambda = \frac{\alpha_i - 1}{x_i^{(K)}}$ to satisfy all the conditions involving $x_i^{(K)}$.

For any j s.t. $x_j^{(K)} = \epsilon$, suppose $x_j^{(k)} < \epsilon$, i.e., x_j falls below ϵ in iteration k and is set to ϵ afterwards. Pick some i s.t. $i \notin T^{(K)}$. After iteration k and $k+1$ respectively, we have:

$$\frac{x_i^{(k)}}{\alpha_i - 1} = \frac{1 - \epsilon \|T^{(k)}\| - \sum_{j' \in T^{(k+1)} \setminus T^{(k)}} x_{j'}^{(k)}}{\sum_{j' \notin T^{(k+1)}} \alpha_{j'} - 1}$$

$$\frac{x_i^{(k+1)}}{\alpha_i - 1} = \frac{1 - \epsilon \|T^{(k+1)}\|}{\sum_{j' \notin T^{(k+1)}} \alpha_{j'} - 1}$$

Algorithm 2 Fast mode-finding of mDir($\boldsymbol{x}; \boldsymbol{\alpha}, \epsilon$)

```
$\langle \alpha_{k_1}, \ldots, \alpha_{k_n} \rangle \leftarrow \langle \alpha_1, \ldots, \alpha_n \rangle$ in ascending order
$s_n \leftarrow \alpha_{k_n} - 1$
for $i = n-1, \ldots, 1$ do
    $s_i = s_{i+1} + \alpha_{k_i} - 1$  ▷ So $s_i = \sum_{j \geq i}(\alpha_{k_j} - 1)$
end for
$t \leftarrow 0$
for $i = 1, \ldots, n$ do
    $x_{k_i} \leftarrow \frac{\alpha_{k_i} - 1}{s_i} \times (1 - \epsilon\, t)$
    if $x_{k_i} < \epsilon$ then
        $x_{k_i} \leftarrow \epsilon$,   $t \leftarrow t + 1$
    end if
end for
return $\langle x_1, \ldots, x_n \rangle$
```

Because for any $j' \in T^{(k+1)} \backslash T^{(k)}$ we have $x_{j'}^{(k)} < \epsilon$, from the two equations above we can deduce that $x_i^{(k)} > x_i^{(k+1)}$, i.e., x_i monotonically decreases over iterations. Therefore,

$$(\alpha_j - 1) \times \frac{x_i^{(K)}}{\alpha_i - 1} < (\alpha_j - 1) \times \frac{x_i^{(k)}}{\alpha_i - 1} = x_j^{(k)} < \epsilon$$

So we get

$$\frac{\alpha_j - 1}{\epsilon} < \frac{\alpha_i - 1}{x_i^{(K)}} = \lambda$$

So we set $\mu_j = \lambda - \frac{\alpha_j - 1}{\epsilon}$ and all the conditions involving $x_j^{(K)}$ are also satisfied. The proof is now complete. □

The worst-case time complexity of Algorithm 1 is $O(n^2)$, but in practice when ϵ is small, the algorithm almost always terminates after only one iteration, leading to linear running time. We also provide a different mode-finding algorithm with better worst-case time complexity $\Theta(n \log n)$ (Algorithm 2). It differs from Algorithm 1 in that the elements of $\boldsymbol{\alpha}$ are first sorted, so we can finish computing $\boldsymbol{x}$ in one pass. It can be more efficient than Algorithm 1 when both ϵ and n are larger. Its correctness can be proved in a similar way to that of Algorithm 1.

2.2 Related Distribution

The closest previous work to mDir is the pseudo-Dirichlet distribution (Larsson and Ugander, 2011). It also allows negative parameters to achieve stronger sparsity. However, the pseudo-Dirichlet distribution is no longer conjugate to the multinomial distribution. Consequently, its maximum a posteriori inference becomes complicated and has no time-complexity guarantee.

3 Learning Mixtures of Gaussians

We first evaluate mDir in learning mixtures of Gaussians from synthetic data. The ground-truth model contains two bivariate Gaussian components with equal mixing probabilities (Figure 5(a)). From the ground-truth we sampled two training datasets of 20 and 200 data points. We then tried to fit a Gaussian mixture model with five components.

Three approaches were tested: maximum likelihood estimation using expectation-maximization (denoted by EM), which has no sparsity preference; mean-field variational Bayesian inference with a Dir prior over the mixing probabilities (denoted by VB-Dir), which is the most frequently used inference approach for Dir with $\alpha < 1$; maximum a posteriori estimation using expectation-maximization with a mDir prior over the mixing probabilities (denoted by EM-mDir). The Dir and mDir priors that we used are both symmetric, i.e., all the elements in vector $\boldsymbol{\alpha}$ have the same value, denoted by α. For mDir, we set $\epsilon = 10^{-5}$. We ran each approach under each parameter setting for 300 times with different random initialization and then reported the average results. During learning, we pruned a Gaussian component whenever its covariance matrix becomes numerically singular (which means the component is estimated from only one or two data samples).

Figure 1–4 show the average test set log likelihood and the effective numbers of mixture components of the models learned with different values of parameter α from 20 and 200 samples respectively. For VB-Dir, we show the results with the α value as low as 10^{-5}. Further decreasing α did not improve the results. It can be seen that both VB-Dir and EM-mDir can achieve better test set likelihood and lower effective numbers of components than EM with proper α values. EM-mDir outperforms VB-Dir even with positive α values, and its performance is further boosted when α becomes negative. The improvement of EM-mDir when α becomes negative is smaller in the 20-sample case than in the 200-sample case. This is because when the train-

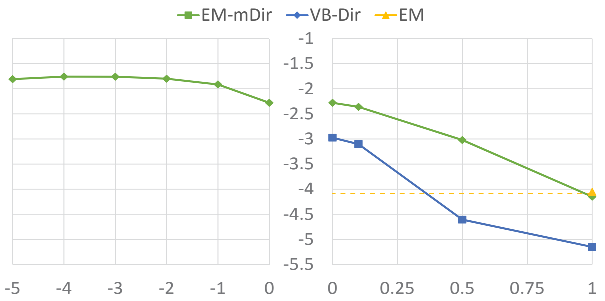


Figure 1: Test set log likelihood vs. the value of α (20 training samples)

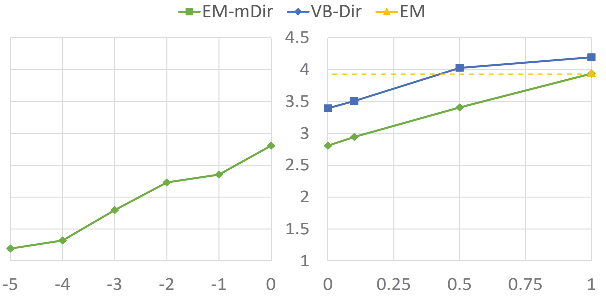


Figure 2: Effective number of components vs. the value of α (20 training samples)

ing dataset is small, a small positive α value may already be sufficient in inducing enough sparsity.

Figure 5(b)–(e) show the typical models learned by VB-Dir and EM-mDir. When the training dataset is small, both Dir and mDir are effective sparsity priors that help prune unnecessary mixture components, though mDir can be more effective with a negative α value. When the training dataset is large, however, the Dir prior is overwhelmed by the likelihood in posterior inference and cannot effectively prune mixture components. On the other hand, with a highly negative α value, mDir is still effective as a sparsity prior.

4 Unsupervised Dependency Parsing

Unsupervised dependency parsing aims to learn a dependency grammar from unannotated text. Previous work has shown that sparsity regularization improves the performance of unsupervised dependency parsing (Johnson et al., 2007; Gillenwater et al., 2010). In our experiments, we tried to learn a dependency model with valence (DMV) (Klein and Manning, 2004) from the Wall Street Journal corpus, with section 2-21 for training and section 23 for testing. Following previous work, we used sentences of length ≤ 10 with punctuation stripped off. Since DMV is an unlexicalized model, the number of dependency rules is small relative to the training corpus size. This suggests that a strong prior can be helpful in counterbalancing the influence of the training data.

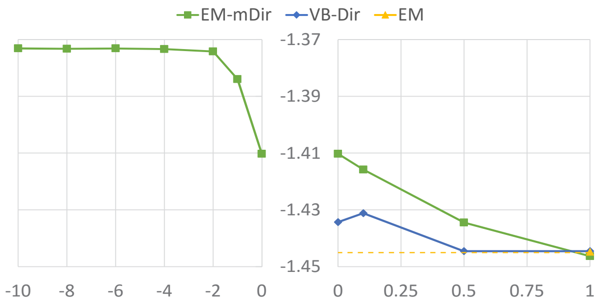


Figure 3: Test set log likelihood vs. the value of α (200 training samples)

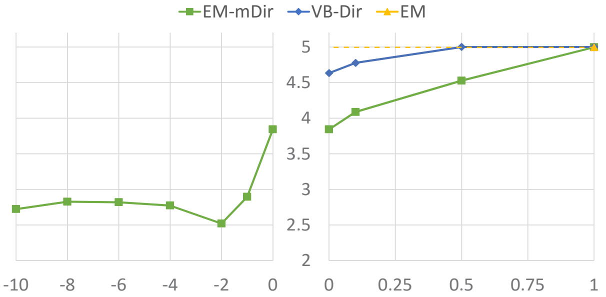
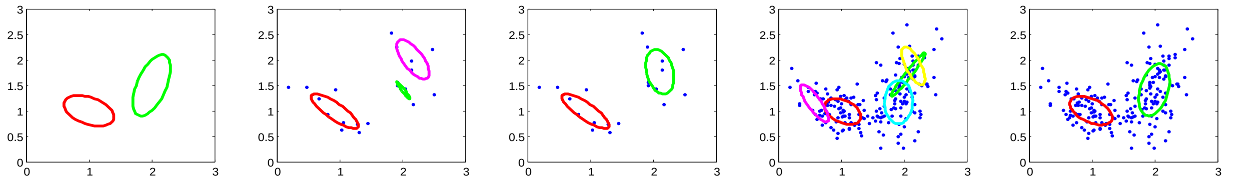


Figure 4: Effective number of components vs. the value of α (200 training samples)

We tested six approaches. With a mDir prior, we tried EM, hard EM, and softmax-EM with $\sigma = 0.5$ (Tu and Honavar, 2012) (denoted by EM-mDir, HEM-mDir, SEM-mDir). With a Dir prior, we tried variational inference, hard variational inference, and softmax variational inference with $\sigma = 0.5$ (Tu and Honavar, 2012) (denoted by VB-Dir, HVB-Dir, SVB-Dir). Again, we used symmetric Dir and mDir priors. For mDir, we set $\epsilon = 10^{-4}$ by default.

Figure 6 shows the directed accuracy of parsing the test corpus using the learned dependency models. It can be seen that with positive α values, Dir and mDir have very similar accuracy under the standard, hard and softmax versions of inference respectively. With negative α values, the accuracy of EM-mDir decreases; but for HEM-mDir and SEM-mDir, the accuracy is significantly improved with moder-



(a) Ground-truth (b) VB-Dir, $\alpha = 10^{-5}$ (c) EM-mDir, $\alpha = -2$ (d) VB-Dir, $\alpha = 10^{-5}$ (e) EM-mDir, $\alpha = -30$

Figure 5: The ground-truth model and four typical models learned by VB-Dir and EM-mDir. (b),(c): 20 training samples. (d),(e): 200 training samples.

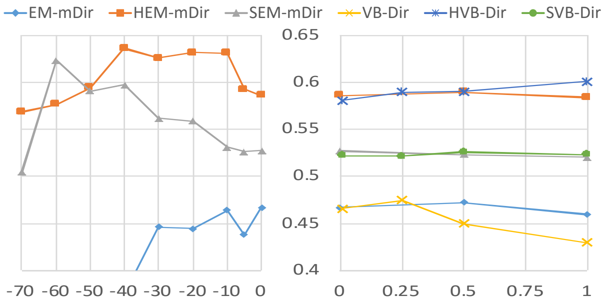


Figure 6: Parsing accuracy vs. the value of α

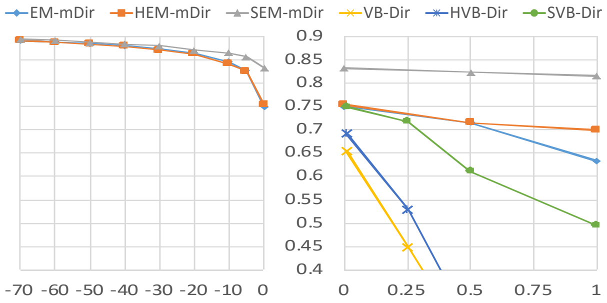


Figure 7: Sparsity of the learned grammars vs. the value of α

ately negative α values. HEM-mDir consistently produces accuracy around 0.63 with a large range of α values (from -10 to -40), which is on a par with the best published results in learning the original DMV model (Cohen and Smith, 2009; Gillenwater et al., 2010; Berg-Kirkpatrick et al., 2010), even though these previous approaches employed more sophisticated features and advanced regularization techniques than ours.

Figure 7 shows the degree of sparsity of the learned dependency grammars. We computed the percentage of dependency rules with probabilities below 10^{-3} to measure the degree of sparsity. It can be seen that even with positive α values, mDir leads to significantly more sparse grammars than Dir does. With negative values of α, mDir can induce even more sparsity.

Figure 8 plots the parsing accuracy with different values of parameter ϵ in mDir (α is set to -20). The best accuracy is achieved when ϵ is neither too large nor too small. This is because if ϵ is too large, the probabilities of dependency rules become too uniform to be discriminative. On the other hand, if ϵ is too small, then the probabilities of many dependency rules may become too small in the early stages of learning and never be able to recover. Similar observation was made by Johnson et al. (2007) when doing maximum a posteriori estimation with a Dir prior (hence with $\epsilon = 0$).

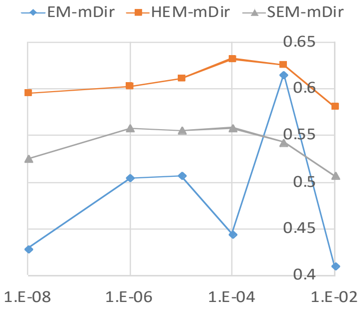


Figure 8: Parsing accuracy vs. the value of ϵ

5 Conclusion

We modify the Dirichlet distribution to allow negative values of parameter $\boldsymbol{\alpha}$ so that it induces stronger sparsity when used as a prior of a multinomial distribution. A second parameter ϵ is introduced which prevents divergence of the normalization factor and also acts as a smoothing factor. Our modified Dirichlet distribution (mDir) is still conjugate to the multinomial distribution. We propose two efficient algorithms for finding the mode of mDir. Our experiments on learning Gaussian mixtures and unsupervised dependency parsing show the advantage of mDir over the Dirichlet distribution.

References

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590. Association for Computational Linguistics.

Shay B. Cohen and Noah A. Smith. 2009. Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction. In *HLT-NAACL*, pages 74–82.

Jennifer Gillenwater, Kuzman Ganchev, João Graça, Fernando Pereira, and Ben Taskar. 2010. Sparsity in dependency grammar induction. In *ACL '10: Proceedings of the ACL 2010 Conference Short Papers*, pages 194–199, Morristown, NJ, USA. Association for Computational Linguistics.

Mark Johnson, Thomas L. Griffiths, and Sharon Goldwater. 2007. Bayesian inference for pcfgs via markov chain monte carlo. In *HLT-NAACL*, pages 139–146.

Dan Klein and Christopher D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of ACL*.

Martin O Larsson and Johan Ugander. 2011. A concave regularization technique for sparse mixture models. In *Advances in Neural Information Processing Systems*, pages 1890–1898.

Kewei Tu and Vasant Honavar. 2012. Unambiguity regularization for unsupervised learning of probabilistic grammars. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1324–1334. Association for Computational Linguistics.