A Proof of Proposition 1

We provide here a detailed proof of Proposition 1.

A.1 Forward Propagation

The optimization problem can be written as

$$\mathsf{csparsemax}(\boldsymbol{z}, \boldsymbol{u}) = \arg\min \quad \tfrac{1}{2}\|\boldsymbol{\alpha}\|^2 - \boldsymbol{z}^\top \boldsymbol{\alpha} \quad \text{s.t.} \quad \begin{cases} \mathbf{1}^\top \boldsymbol{\alpha} = 1 \\ \mathbf{0} \leq \boldsymbol{\alpha} \leq \boldsymbol{u}. \end{cases}$$

The Lagrangian function is:

$$\mathcal{L}(\boldsymbol{\alpha}, \tau, \boldsymbol{\mu}, \boldsymbol{\nu}) = -\frac{1}{2}\|\boldsymbol{\alpha}\|^2 - \boldsymbol{z}^\top \boldsymbol{\alpha} + \tau(\mathbf{1}^\top \boldsymbol{\alpha} - 1) - \boldsymbol{\mu}^\top \boldsymbol{\alpha} + \boldsymbol{\nu}^\top (\boldsymbol{\alpha} - \boldsymbol{u}). \tag{9}$$

To obtain the solution, we invoke the Karush-Kuhn-Tucker conditions. From the stationarity condition, we have $\mathbf{0} = \boldsymbol{\alpha} - \boldsymbol{z} + \tau \mathbf{1} - \boldsymbol{\mu} + \boldsymbol{\nu}$, which due to the primal feasibility condition implies that the solution is of the form:

$$\boldsymbol{\alpha} = \boldsymbol{z} - \tau \mathbf{1} + \boldsymbol{\mu} - \boldsymbol{\nu}. \tag{10}$$

From the complementarity slackness condition, we have that $0 < \alpha_j < u_j$ implies that $\mu_j = \nu_j = 0$ and therefore $\alpha_j = z_j - \tau$. On the other hand, $\mu_j > 0$ implies $\alpha_j = 0$, and $\nu_j > 0$ implies $\alpha_j = u_j$. Hence the solution can be written as $\alpha_j = \max\{0, \min\{u_j, z_j - \tau\}$, where τ is determined such that the distribution normalizes:

$$\tau = \frac{\sum_{j \in \mathcal{A}} z_j + \sum_{j \in \mathcal{A}_R} u_j - 1}{|\mathcal{A}|}, \tag{11}$$

with $\mathcal{A} = \{j \in [J] \mid 0 < \alpha_j < u_j\}$ and $\mathcal{A}_R = \{j \in [J] \mid \alpha_j = u_j\}$. Note that τ depends itself on the set $\mathcal{A}$, a function of the solution. In §A.3, we describe an algorithm that searches the value of τ efficiently.

A.2 Gradient Backpropagation

We now turn to the problem of backpropagating the gradients through the constrained sparsemax transformation. For that, we need to compute its Jacobian matrix, i.e., the derivatives $\frac{\partial \alpha_i}{\partial z_j}$ and $\frac{\partial \alpha_i}{\partial u_j}$ for $i, j \in [J]$. Let us first express $\boldsymbol{\alpha}$ as

$$\alpha_i = \begin{cases} 0, & i \in \mathcal{A}_L, \\ z_i - \tau, & i \in \mathcal{A}, \\ u_i, & i \in \mathcal{A}_R, \end{cases} \tag{12}$$

with τ as in Eq. 11. Note that we have $\partial\tau/\partial z_j = \mathbb{1}(j \in \mathcal{A})/|\mathcal{A}|$ and $\partial\tau/\partial u_j = \mathbb{1}(j \in \mathcal{A}_R)/|\mathcal{A}|$. Thus, we have the following:

$$\frac{\partial \alpha_i}{\partial z_j} = \begin{cases} 1 - 1/|\mathcal{A}|, & \text{if } j \in \mathcal{A} \text{ and } i = j \\ -1/|\mathcal{A}|, & \text{if } i, j \in \mathcal{A} \text{ and } i \neq j \\ 0, & \text{otherwise,} \end{cases} \tag{13}$$

and

$$\frac{\partial \alpha_i}{\partial u_j} = \begin{cases} 1, & \text{if } j \in \mathcal{A}_R \text{ and } i = j \\ -1/|\mathcal{A}|, & \text{if } j \in \mathcal{A}_R \text{ and } i \in \mathcal{A} \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

Finally, we obtain:

$$\begin{aligned} \mathrm{d}z_j &= \sum_i \frac{\partial \alpha_i}{\partial z_j} \mathrm{d}\alpha_i \\ &= \mathbb{1}(j \in \mathcal{A}) \left(\mathrm{d}\alpha_j - \frac{\sum_{i \in \mathcal{A}} \mathrm{d}\alpha_i}{|\mathcal{A}|} \right) \\ &= \mathbb{1}(j \in \mathcal{A})(\mathrm{d}\alpha_j - m), \end{aligned} \tag{15}$$

Algorithm 1 Pardalos and Kovoor's Algorithm

```
1: input: a, b, c, d
2: Initialize working set W ← {1, ..., J}
3: Initialize set of split points:
       P ← {a_j, b_j}_{j=1}^J ∪ {±∞}
4: Initialize τ_L ← −∞, τ_R ← ∞, s_tight ← 0, ξ ← 0.
5: while W ≠ ∅ do
6:    Compute τ ← Median(P)
7:    Set s ← s_tight + Σ_{j∈W | b_i<τ} c_j b_j + Σ_{j∈W | a_j>τ} c_j a_j + (ξ + Σ_{j∈W | a_j≤τ≤b_j} c_j)τ
8:    If s ≤ d, set τ_L ← τ; if s ≥ d, set τ_R ← τ
9:    Reduce set of split points: P ← P ∩ [τ_L, τ_R]
10:   Update tight-sum: s_tight ← s_tight + Σ_{j∈W | b_i<τ_L} c_j b_j + Σ_{j∈W | a_j>τ_R} c_j a_j
11:   Update slack-sum: ξ ← ξ + Σ_{j∈W | a_j≤τ_L ∧ b_j≥τ_R} c_j
12:   Update working set: W ← {j ∈ W | τ_L < a_j < τ_R ∨ τ_L < b_j < τ_R}
13: end while
14: Define y* ← (d − s_tight)/ξ
15: Set x*_j = max{a_j, min{b_j, y}}, ∀j ∈ [J]
16: output: x*.
```

and

$$\begin{aligned} \mathrm{d}u_j &= \sum_i \frac{\partial \alpha_i}{\partial u_j}\mathrm{d}\alpha_i \\ &= \mathbb{1}(j \in \mathcal{A}_R)\left(\mathrm{d}\alpha_j - \frac{\sum_{i\in\mathcal{A}}\mathrm{d}\alpha_i}{|\mathcal{A}|}\right) \\ &= \mathbb{1}(j \in \mathcal{A}_R)(\mathrm{d}\alpha_j - m), \end{aligned} \tag{16}$$

where $m = \frac{1}{|\mathcal{A}|}\sum_{j\in\mathcal{A}}\mathrm{d}\alpha_j$.

A.3 Linear-Time Evaluation

Finally, we present an algorithm to solve the problem in Eq. 6 in linear time.

Pardalos and Kovoor (1990) describe an algorithm, reproduced here as Algorithm 1, for solving a class of singly-constrained convex quadratic problems, which can be written in the form above (where each $c_j \geq 0$):

$$\begin{aligned} &\text{minimize} \sum_{j=1}^{J} c_j x_j^2 \\ &\text{s.t.} \ \textstyle\sum_{j=1}^{J} c_j x_j = d, \\ &\qquad a_j \leq x_j \leq b_j, \quad j = 1, \ldots, J. \end{aligned} \tag{17}$$

The solution of the problem in Eq. 17 is of the form $x_j^\star = \max\{a_j, \min\{b_j, y\}\}$, where $y \in [a_j, b_j]$ is a constant. The algorithm searches the value of this constant (which is similar to τ in our problem), which lies in a particular interval of split-points (line 3), iteratively shrinking this interval. The algorithm requires computing medians as a subroutine, which can be done in linear time (Blum et al., 1973). The overall complexity in $O(J)$ (Pardalos and Kovoor, 1990). The same algorithm has been used in NLP by Almeida and Martins (2013) for a budgeted summarization problem.

To show that this algorithm applies to the problem of evaluating csparsemax, it suffices to show that our problem in Eq. 6 can be rewritten in the form of Eq. 17. This is indeed the case, if we set:

$$x_j = \frac{p_j - z_j}{2} \tag{18}$$

$$a_j = -z_j/2 \tag{19}$$

$$b_j = (u_j - z_j)/2 \tag{20}$$

$$c_j = 1 \tag{21}$$

$$d = \frac{1 - \sum_{j=1}^{J} z_j}{2}. \tag{22}$$

B Examples of Translations

We show some examples of translations obtained for the German-English language pair with different systems. *Blue* highlights the parts of the reference that are correct and **red** highlights the corresponding problematic parts of translations, including repetitions, dropped words or mistranslations.

input	überlassen sie das ruhig uns .
reference	*leave that up to us* .
softmax	**give us a silence** .
sparsemax	leave that to us .
csoftmax	**let's** leave that .
csparsemax	leave it to us .

input	so ungefähr , sie wissen schon .
reference	*like that* , *you know* .
softmax	**so , you know , you know** .
sparsemax	**so , you know , you know** .
csoftmax	**so , you know , you know** .
csparsemax	like that , you know .

input	und wir benutzen dieses wort mit solcher verachtung .
reference	and we say that word *with such contempt* .
softmax	and we use this word with such **contempt contempt** .
sparsemax	and we use this word with such contempt .
csoftmax	and we use this word with **like this** .
csparsemax	and we use this word with such contempt .

input	wir sehen das dazu , dass phosphor wirklich kritisch ist .
reference	we can see *that* phosphorus is really critical .
softmax	we see **that that** phosphorus is really critical .
sparsemax	we see **that that** phosphorus really is critical .
csoftmax	we see **that that** phosphorus is really critical .
csparsemax	we see that phosphorus is really critical .

input	also müssen sie auch nicht auf klassische musik verzichten , weil sie kein instrument spielen .
reference	so *you don't need to abstain from listening to* classical music because *you don't play* an instrument .
softmax	so you don't have to **rely on** classical music because you don't **have** an instrument .
sparsemax	so **they** don't have to **kill** classical music because **they** don't play an instrument .
csoftmax	so **they** don't have to **rely on** classical music , because **they** don't play an instrument .
csparsemax	so you don't have to get **rid of** classical music , because you don't play an instrument .

input	je mehr ich aber darber nachdachte , desto mehr kam ich zu der ansicht , das der fisch etwas weiß .
reference	the more i thought about it , however , the more *i came to the view that this fish knows something* .
softmax	the more i thought about it , the more **i got to the point of the fish** .
sparsemax	the more i thought about it , the more **i got to the point of view of the fish** .
csoftmax	but the more i thought about it , the more **i came to mind , the fish** .
csparsemax	the more i thought about it , the more i **came to the point that the fish knows** .

input	all diese menschen lehren uns , dass es noch andere existenzmöglichkeiten , andere denkweisen , andere wege zur orientierung auf der erde gibt .
reference	all of these peoples teach us that there are *other ways of being , other ways of thinking , other ways of orienting yourself in the earth .*
softmax	all of these people teach us that there are **others , other ways , other ways of guidance to the earth .**
sparsemax	all these people are teaching us that there are other **options , other ways , different** ways of guidance on earth .
csoftmax	all of these people teach us that there's other ways of **doing** other ways of thinking , other ways of guidance on earth .
csparsemax	all these people teach us that there are other **actors** , other ways of thinking , other ways of guidance on earth .

input	in der reichen welt , in der oberen milliarde , könnten wir wohl abstriche machen und weniger nutzen , aber im durchschnitt wird diese zahl jedes jahr steigen und sich somit insgesamt mehr als verdoppeln , die zahl der dienste die pro person bereitgestellt werden .
reference	in the rich world , perhaps the top one billion , we probably *could cut back and use less* , but every year , this number , on average , is going to go up , *and so* , over all , that will *more than double the services delivered per person* .
softmax	in the rich world , in the upper billion , we might **be able to do** and **use** less **use** , but on average , that number is going to increase every year and so on , **which is** the number of services **that are being put in .**
sparsemax	in the rich world , in the upper billion , we may **be able to do** and **use** less **use** , but in average , that number is going to rise every year , and so **much** more than double , the number of services that are being **put together** .
csoftmax	in the rich world , in the upper billion , we might **be able to take off** and use less , but in average , this number is going to increase every year and so **on** , and that's the **number of people who** are being put together per person .
csparsemax	in the rich world , in the upper billion , we may **be able to turn off** and use less , but in average , that number will rise every year and so far more than double , the number of services that are being put into a person .