Philipp Lintl
12152498
lintl.philipp@gmail.com

1 A K-Sided Die

We will derive the MAP solution for the $\boldsymbol{\theta}$ parameters of a Dirichlet multinomial model which models the outcome of a K-sided Die.
Assume we have a K-sided die and we observed N dice rolls $\{x_1, \ldots, x_N\}$, where x_i denotes the result from the ith roll, e.g $x_i \in \{1, \ldots, K\}$. Assumind iid data, likelihood is

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{n=1}^{N} \theta_{x_n} = \prod_{k=1}^{K} \theta_k^{N_k}, \quad N_k = \sum_{n=1}^{N} [x_n = k],$$

where $[c] = 1$ if the condition c holds and $[c] = 0$ otherwise.
Because of multinomial, the parameter has the constraint:

$$\sum_{k=1}^{K} \boldsymbol{\theta}_k = 1, \quad \forall k: \ \theta_k \geq 0$$

If we wish to introduce a Dirichlet prior on $\boldsymbol{\theta}$, it is a distribution over vectors, as well as a conjugate prior to our likelihood. The prioir is given by:

$$Dir(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1}, \quad for \ \boldsymbol{\theta} \in S_K,$$

where S_K is the set of the K-dimensional vectors that satisfy the constraints of the multinomial. Multiplying the likelihood with the prior and some manipulations we end up with the posterior as follows:

$$\begin{aligned} p(\boldsymbol{\theta}|\mathcal{D}) &\propto p(\mathcal{D}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \\ &\propto \prod_{k=1}^{K} \theta_k^{N_k} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} = \prod_{k=1}^{K} \theta_k^{N_k + \alpha_k - 1} \end{aligned}$$

From this we can identify a Dirichlet Distribution, hence

$$p(\boldsymbol{\theta}|\mathcal{D}) = Dir(\boldsymbol{\theta}|\alpha_1 + N_1, \ldots, \alpha_K + N_K).$$

Now, find the MAP estimate $\boldsymbol{\theta}_{MAP}$ of $\boldsymbol{\theta}$. Dont forget to use the constraints on $\boldsymbol{\theta}$. Your answer should consist of the folowing steps:

1.1

Derive the log posterior:
As it is a Dirichlet we arrive at:

$$\ln p(\boldsymbol{\theta}|\mathcal{D}) = -\ln B(\boldsymbol{\alpha} + \mathbf{N}) \sum_{k=1}^{K} (N_k + \alpha_k - 1) \cdot \ln \theta_k$$

The normalizing factor of the Dirichlet $\frac{1}{B(\boldsymbol{\alpha}+\boldsymbol{N})}$ (whereas $\boldsymbol{\alpha} + \boldsymbol{N}$ is the vector form for $\alpha_1 + N_1, \ldots, \alpha_K + N_K$) can be disregarded, as with the ln applied, it would be simply an added term independent of $\boldsymbol{\theta}$ which would just be 0 for the derivatives wrt to that parameter. It is mentioned nonetheless in the equation.

1.2

Define the Lagrangian $l(\boldsymbol{\theta}, \lambda, \boldsymbol{\mu})$, where λ is the L multiplier that corresponds to the sum-to-one constraint and μ_k the L multiplier that corresponds to $\theta_k \geq 0$ constraint. Note that although it is not strictly necessery to include the second constraint for this problem, you are **required** to include both for the assignment.

We have two constraints:

$$\begin{aligned} \sum_{k=1}^{K} \theta_k &= 1 \\ \theta_k &\geq 0, \quad \forall k \in \{1, \dots, K\} \end{aligned}$$

The Lagrangian thus emerges as:

$$\begin{aligned} l(\boldsymbol{\theta}, \lambda, \boldsymbol{\mu}) &= \ln p(\boldsymbol{\theta}|\mathcal{D}) + \lambda \cdot (\sum_{k=1}^{K} \theta_k - 1) + \sum_{k=1}^{K} \mu_k \cdot \theta_k) \\ &= -\ln B(\boldsymbol{\alpha} + \mathbf{N}) + \sum_{k=1}^{K} (N_k + \alpha_k - 1) \cdot \ln \theta_k + \lambda \cdot \left(\sum_{k=1}^{K} \theta_k - 1 \right) + \sum_{k=1}^{K} \mu_k \cdot \theta_k \end{aligned}$$

1.3

State the KKT conditions.

Referring to the practical handin, the derivative of the above stated primal Lagrangian are not part of the KKT conditions. Thus in accordance with the slides and Bishop Appendix E we have all the equality, inequality and the constraints on the inequality multiplier:

$$\begin{aligned} \sum_{k=1}^{K} \theta_k &= 1 \\ \theta_k &\geq 0, \quad \forall k \in \{1, \dots, K\} \\ \mu_k &\geq 0, \quad \forall k \in \{1, \dots, K\} \\ \mu_k \cdot \theta_k &= 0, \quad \forall k \in \{1, \dots, K\} \end{aligned}$$

1.4

Find $\boldsymbol{\theta}_{MAP}$.

We achieve that by deriving the lagrangian for the parameters $\boldsymbol{\theta}$ and setting it to zero. Individual derivations for each θ_k are given by:

$$\frac{\partial l(\boldsymbol{\theta}, \lambda, \boldsymbol{\mu})}{\partial \theta_k} = \frac{N_k + \alpha_k - 1}{\theta_k} + \lambda + \mu_k \stackrel{!}{=} 0 \tag{1}$$

$$\text{(rearranging)} \quad \Rightarrow \quad N_k + \alpha_k - 1 = -\theta_k \cdot \lambda - \underbrace{\theta_k \cdot \mu_k}_{=0} \tag{2}$$

$$\Rightarrow \quad \frac{1 - N_k - \alpha_k}{\lambda} = \theta_k \tag{3}$$

Now, we need to find λ to plugg it into the appropriate position.
Assuming we take the derivative wrt to all θ_k, similarly to the practical, we can take the sum over all those

equations as in (1):

$$\sum_{k=1}^{K}\left(\frac{N_k+\alpha_k-1}{\theta_k}+\lambda+\mu_k\right)=0$$

$$(\cdot\theta_k \text{ allowed as} \theta_k \geq 0) \quad \sum_{k=1}^{K}\left(N_k+\alpha_k-1+\theta_k\cdot\lambda+\underbrace{\theta_k\cdot\mu_k}_{=0}\right)=0$$

$$-\sum_{k=1}^{K} N_k+\alpha_k-1=\lambda\cdot\underbrace{\sum_{k=1}^{K}\theta_k}_{=1}$$

$$\lambda=\sum_{k=1}^{K} 1-N_k-\alpha_k$$

We plugg this into the equation derived above:

$$\Rightarrow \quad \theta_k=\frac{1-N_k-\alpha_k}{\sum_{k=1}^{K}1-N_k-\alpha_k}$$

The entire $\boldsymbol{\theta}$ vector thus is given by:

$$\boldsymbol{\theta}_{MAP}=\frac{1-(\boldsymbol{\alpha}+\boldsymbol{N})}{\sum_{k=1}^{K}1-N_k-\alpha_k},$$

again $\boldsymbol{alpha}+\boldsymbol{N}=(\alpha_1+N_1,\ldots,\alpha_K+N_K)$ is this vector.

2 Maximum Margin Classifier

Assume a dataset $\mathcal{D}=\{(x_1,t_1)\ldots,(x_N,t_N)\}$ where $x_n\in\mathbb{R}^2$ and $t_n\in\{-1,+1\}$. Upon inspection we suspect that there exists a circle with radius $\mathcal{R}$ that separates the data (up to some exceptions). The datapoints within the circle are assigned label $t_n=-1$ and the datapoints outside of the circle are assigned label $t_n=+1$. Now, we do not want to find any circle that separates the data, we want to find the circle with radius R that has the maximum margin. For this assignment, we will make the simplifying assumption that the data (and thus the circle that separates the data) lies centered around the origin (0, 0). Under the assumption that the circle perfectly separates the two classes, the distance between the decision boundary and any datapoint x_n is given by

$$t_n(\|\mathbf{x}_n\|-R)=\frac{t_n(\alpha\|\mathbf{x}_n\|-\mathcal{R})}{\alpha},$$

where $R=\alpha\mathcal{R}$ and $\alpha>0$. Note that, the distance to the decision boundary is invariant to the rescaling $\alpha\rightarrow\|\alpha$. We can use this to set

$$t_n(\|\mathbf{x}_n\|-\mathcal{R})=1$$

for the point x_n that is closest to the decision boundary. As such, the constraint

$$t_n(\|\mathbf{x}_n\|-\mathcal{R})\geq 1, \quad \forall n=1,\ldots,N$$

holds for all datapoints. However, based on Figure 1, it does not seem that the data is perfectly separable, hence, we will introduce slack variables. We will now state the primal program that will fi
nd such a circle:

$$\arg\min_{\mathcal{R},\alpha,\xi}\frac{1}{2}\alpha^2+C\sum_{n=1}^{N}\xi_n \quad \textbf{s.t.}\forall n: t_n(\alpha\|\mathbf{x}_n\|-R)\geq 1-\xi_n,\xi_n\geq 0$$

2.1

Introduce Lagrange multipliers for the constraints and write down the primal Lagrangian. Use the following notation: $\{\lambda_n\}$ are the Lagrange multipliers for the first constraing and $\{\mu_n\}$ for the second.

$$\mathcal{L}(R, \boldsymbol{\xi}, \alpha, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \frac{1}{2}\alpha^2 + C\sum_{n=1}^{N}\xi_n - \sum_{n=1}^{N}\lambda_n \cdot \{t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n\} - \sum_{n=1}^{N}\mu_n \cdot \xi_n$$

2.2

How many KKT conditions are there? Write down all KKT conditions.
Similar to the practical, KKT conditions are composed of the following:

$$\underbrace{\begin{aligned}&\lambda_n \geq 0\\ &t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n \geq 0\\ &\lambda_n\{t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n\} = 0\end{aligned}}_{3N} \qquad \underbrace{\begin{aligned}&\mu_n \geq 0 \quad \forall n \in \{1, \ldots, N\}\\ &\xi_n \geq 0 \quad \forall n \in \{1, \ldots, N\}\\ &\mu_n\xi_n = 0 \quad \forall n \in \{1, \ldots, N\}\end{aligned}}_{3N}$$

So in total there are 6N KKT conditions.

2.3

Derive the dual Lagrangian and specify the dual optimization problem. That is, eliminate the primal variables $\{\alpha, R, \xi_1, \ldots, \xi_N\}$ using the $\frac{\partial l}{\partial \rho} = 0$ equations, where l is the primal Lagrangian and ρ is a primal variable. Do not forget to specify the constraints on the remaining dual variables.

First the described equations:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial R} &= \sum_{n=1}^{N}\lambda_n t_n \stackrel{!}{=} 0\\ \frac{\partial \mathcal{L}}{\partial \alpha} &= \alpha - \sum_{n=1}^{N}\lambda_n t_n \|x_n\| \stackrel{!}{=} 0\\ \Rightarrow \quad \alpha &= \sum_{n=1}^{N}\lambda_n t_n \|\mathbf{x}_n\|\\ \frac{\partial \mathcal{L}}{\partial \xi_n} &= C - \lambda_n - \mu_n \stackrel{!}{=} 0 \quad \forall n\end{aligned}$$

These will help us to eliminate the primal variables.

$$
\begin{aligned}
\mathcal{L}(R, \boldsymbol{\xi}, \alpha, \boldsymbol{\mu}, \boldsymbol{\lambda}) &= \frac{1}{2}\alpha^2 + C\sum_{n=1}^{N}\xi_n - \sum_{n=1}^{N}\lambda_n \cdot \{t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n\} - \sum_{n=1}^{N}\mu_n \cdot \xi_n \\
&= \frac{1}{2}\alpha^2 + C\sum_{n=1}^{N}\xi_n - \alpha\underbrace{\sum_{n=1}^{N}\lambda_n t_n\|\mathbf{x}_n\|}_{=\alpha} + R\underbrace{\sum_{n=1}^{N}\lambda_n t_n}_{0} + \sum_{n=1}^{N}\lambda_n - \sum_{n=1}^{N}\lambda_n\xi_n - \sum_{n=1}^{N}\mu_n \cdot \xi_n \\
&= \frac{1}{2}\alpha^2 + \sum_{n=1}^{N}\xi_n\underbrace{(C - \lambda_n - \mu_n)}_{=0} - \alpha^2 + \sum_{n=1}^{N}\lambda_n \\
&= -\frac{1}{2}\alpha^2 + \sum_{n=1}^{N}\lambda_n \\
&= -\frac{1}{2}\left(\sum_{n=1}^{N}\lambda_n t_n\|\mathbf{x}_n\|\right)^2 + \sum_{n=1}^{N}\lambda_n \\
\text{(squares of sums rule)} \quad &= \sum_{n=1}^{N}\lambda_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{M}\lambda_n\lambda_m t_n t_m\|\mathbf{x}_n\|\|\mathbf{x}_m\| \\
&= \sum_{n=1}^{N}\lambda_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{M}\lambda_n\lambda_m t_n t_m K_{nm},
\end{aligned}
$$

The remaining constraints are:

$$\lambda_n \in [0, C] \quad \forall n$$

Which is a short form for

$$
\begin{aligned}
\lambda_n &\geq 0, \forall n \\
\mu_n &\geq 0, \forall n \\
C &= \lambda_n + \mu_n, \forall n
\end{aligned}
$$

The dual program thus is

$$\arg\min_{\boldsymbol{\lambda}} = \sum_{n=1}^{N}\lambda_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{M}\lambda_n\lambda_m t_n t_m K_{nm}$$

2.4

Whereas $K_{nm} = K(\mathbf{x}_n, \mathbf{x}_m) = \|\mathbf{x}_n\|\|\mathbf{x}_m\| = \sqrt{x_{n1}^2 + x_{n2}^2}\sqrt{x_{m1}^2 + x_{m2}^2}$

2.5

The dual program will return optimal values for $\{\lambda_n\}$. What is the minimum number of $\{\lambda_n\}$ for which $0 < \lambda_n < C$ will hold? Explain your answer

If $0 < \lambda_n$ then according to one of the KKT conditions $t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n = 0$ must hold. Also if $\lambda_n < C$ then $\mu_n > 0$, also then $\xi_n = 0$.
Therefore, when $0 < \lambda_n < C$, $t_n(\alpha\|x_n\| - R) = 1$- This is true for the closest point to the decision boundary. Thats the reason, this statement is true , exactly for that one point closest to the boundary.

2.6

2.7

Use the KKT conditions to derive wich data cases x_n will have $\lambda_n > 0$ and which ones will have $\mu_n > 0$.

- datapoint inside circle: $t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n > 0$, $\Rightarrow \lambda_n = 0 \Rightarrow \mu_n = c$

- Outside the circle: $\xi_n > 0 \Rightarrow x_n = 0 \Rightarrow \lambda_n = C$
- Inside or on circle: $\xi = 0 \Rightarrow \mu_n \geq 0$
- Outside or on circle: $t_n(\alpha\|\mathbf{x}_n\| - R) - 1 + \xi_n = 0 \Rightarrow \lambda_n \geq 0$

2.8

2.9

If we would use a Radial Basis Function (RBF) kernel instead of K(,) as de
fined in by you in question 4, can the decision boundary be different from a circle in x-space? If yes, describe geometrically what kind of solutions we may expect when using an RBF kernel
Yes it can be different from a circle. As stated in the practical: An RBF kernel will result in flexible and possibly disjunct decision boundaries. Additionally, as shown in the lecture, the RBF Kernel implies a projection into infinitely high dimensional space which according to Covers theorem is favourable as: a non linearly separable training data set can with high probability be transformed into a training set that is linearly separatable by projecting it into a higher-dimensional space (Covers Theorem). Geometrically it can have shapes around certain data points or even regions.