Chapter 5: Loss functions

Problem 5.1

Show that the logistic sigmoid function sig[z] maps $z = -\infty$ to 0, $z = 0$ to 0.5, and $z = \infty$ to 1, where:

$$\text{sig}[z] = \frac{1}{1+\exp[-z]}$$

When $z = -\infty$:

$$\text{sig}[-\infty] = \frac{1}{1+\exp[\infty]} = \frac{1}{1+\exp[\infty]} = \frac{1}{\infty} = 0$$

When $z = 0$:

$$\text{sig}[-\infty] = \frac{1}{1+\exp[0]} = \frac{1}{1+\exp[0]} = \frac{1}{1+1} = \frac{1}{2}$$

When $z = \infty$:

$$\text{sig}[\infty] = \frac{1}{1+\exp[-\infty]} = \frac{1}{1+\exp[-\infty]} = \frac{1}{1+0} = 1$$

Problem 5.2

The loss L for binary classification for a single training pair $\{\mathbf{x}, y\}$ is:

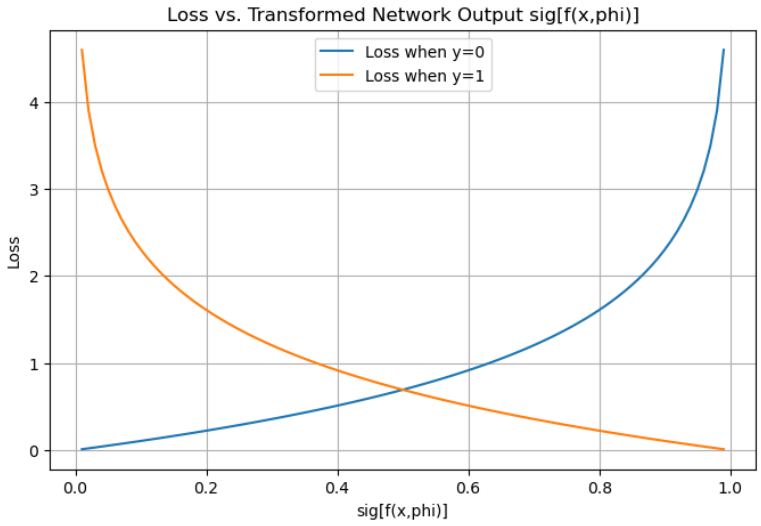
$$L = -1(1-y)\log\left[1-\text{sig}[f[\mathbf{x},\phi]]\right] - y\log\left[\text{sig}[f[\mathbf{x},\phi]]\right]$$

where sig[$\bullet$] is defined above. Plot this loss as a function of the transformed network output sig$[f[\mathbf{x},\phi]] \in [0,1]$ when the training label $y = 0$ and when $y = 1$.

Problem 5.3

Suppose we want to build a network that predicts the direction y in radians of the prevailing wind based on local measurements of barometric pressure $\curvearrowleft$. A suitable distribution over circular domains is the von Mises distribution (figure 5.13 from book)

$$P(y|\mu,\kappa) = \frac{\exp[\kappa\cos(y-\mu)}{2\pi \cdot \text{Bessel}_0[\kappa]}$$



where μ is a measure of the mean direction and κ is a measure of the concentration (i.e., the inverse of the variance). The term $\text{Bessel}_0[\kappa]$ is a modified Bessel function of order 0. Use the recipe from section 5.2 to develop a loss function for learning the parameter μ of a model $f[\mathbf{x}, \phi]$ to predict the most likely wind direction. your solution should treat the concentration κ as constant. How would you perform inference?

To train the model, find the network paramaters $\hat{\phi}$ that minimise the negative log-likelihood (NLL) function over the training set. The NLL for the above distribution can be written as:

$$\begin{aligned} L &= \sum_{i=1}^{I} -\log\left[\frac{\exp[\kappa\cos(y_i - f[\mathbf{x}_i, \phi]]}{2\pi \cdot \text{Bessel}_0[\kappa]}\right] \\ &= -\kappa\cos(y_i - f[\mathbf{x}_i, \phi]) + \log[\text{Bessel}_0[\kappa]] \end{aligned}$$

Since κ is constant, the above loss function can be simplified to:

$$L = -\cos(y_i - f[\mathbf{x}_i, \phi])$$

To perform inference you would take the maximum of the distribution which is just the predicted parameter μ. This may be out of the range $[-\pi, \pi]$, in which case we would add/remove multiples of 2π until it is in the original range.

Problem 5.4

Sometimes the outputs y for input $\mathbf{x}$ are multimodal; there is more than one valid prediction for a given input. Here, we might use a weighted sum of normal components as the distribution over the output. This is known as a *mixture of Gaussians* model. For example, a mixture of two Gaussians has parameters $\boldsymbol{\theta} = \{\lambda, \mu_1, \sigma_1^2, \mu_2, \sigma_2^2$:

$$P(y|\lambda,\mu_1,\sigma_1^2,\mu_2,\sigma_2^2) = \frac{\lambda}{\sqrt{2\pi\sigma_1^2}}\exp\left[\frac{-(y-\mu_1)^2}{2\sigma_1^2}\right] + \frac{1-\lambda}{\sqrt{2\pi\sigma_2^2}}\exp\left[\frac{-(y-\mu_2)^2}{2\sigma_2^2}\right]$$

where $\lambda \in [0,1]$ controls the relative weight of the two components, which have means μ_1, μ_2 and variances σ_1^2, σ_2^2, respectively. This model can represent a distribution with two peaks or a distribution with one peak but a more complex shape.

Use the recipe from section 5.1 to construct a loss function for training a model $\mathbf{f}[\mathbf{x}, \phi]$ that takes input x, has parameter ϕ, and predicts a mixture of two Gaussians. The loss should be based on I training data pairs $\{x_i, y_i\}$. What problems do you foresee when performing inference?

The NLL for the above distribution can be written as, assuming $\mathbf{f}_d[\mathbf{x}_i, \phi]$ is the d^{th} set of network outputs:

$$L = \sum_{i=1}^{I} -\log\left[\frac{\text{sig}[\mathbf{f}_1[\mathbf{x}_i,\phi]]}{\sqrt{2\pi\mathbf{f}_3[\mathbf{x}_i,\phi]^2}}\exp\left[\frac{-(y_i-\mathbf{f}_2[\mathbf{x}_i,\phi])^2}{2\mathbf{f}_3[\mathbf{x}_i,\phi]^2}\right] + \frac{1-\text{sig}[\mathbf{f}_1[\mathbf{x}_i,\phi]]}{\sqrt{2\pi\mathbf{f}_4[\mathbf{x}_i,\phi]^2}}\exp\left[\frac{-(y_i-\mathbf{f}_4[\mathbf{x}_i,\phi])^2}{2\mathbf{f}_4[\mathbf{x}_i,\phi]^2}\right]\right]$$

Inference is slightly more tricky as there is no simple closed form expression for the mode of this distribution. We might have to find it using optimization

Problem 5.5

Consider extending the model from problem 5.3 to predict the wind direction using a mixture of two von Mises distributions. Write an expression for the likelihood $P(y|\boldsymbol{\theta})$ for this model. How many outputs will the network need to produce?

A mixture of two von Mises distributions can be written as follows:

$$P(y|\mu,\kappa) = \pi \cdot \frac{\exp[\kappa\cos(y-\mu_1)}{2\pi \cdot \text{Bessel}_0[\kappa]} + (1-\pi)\cdot\frac{\exp[\kappa\cos(y-\mu_2)}{2\pi \cdot \text{Bessel}_0[\kappa]}$$

The network would need to produce three outputs: the mean for the first von Mises distribution (μ_1), the mean for the second von Mises distribution (μ_2), and the mixing coefficient π.

Problem 5.6

Consider building a model to predict the number of pedestrians $y \in \{0, 1, 2, \dots\}$ that will pass a given point in the city in the next minute, based on data x that contains information about the time of day, the longitude and latitude, and the type of neighborhood. A suitable distribution for modeling counts is the Poisson distribution (figure 5.15 from book). This has a single parameter $\lambda > 0$ called the rate that represents the mean of the distribution.

The distribution has probability density function:

$$P(y=k)=\frac{\lambda^k e^{-\lambda}}{k!}$$

Design a loss function for this model assuming we have access to I training pairs $\{\mathbf{x}_i, y_i\}$.

In this context λ is not fixed, but a function of the data: $\lambda = f[\mathbf{x}, \phi]$. The NLL function for a single observation $(\mathbf{x}, y)$ is given by:

$$-\log\left(\frac{\lambda^k e^{-\lambda}}{k!}\right) = -k\log(\lambda) + \lambda + \log(k!)$$

Since $k!$ is a constant for all true counts k, it can be ignored. To find the loss over all I training pairs, take the sum of the negative log-likelihoods for all observations:

$$L = \sum_{i=1}^{I}[-y_i\log(\lambda_i) + \lambda_i]$$

Problem 5.7

Consider a multivariate regression problem where we predict 10 outputs so $\mathbf{y} \in \mathbb{R}^{10}$, and model each with an independent normal distribution where the means μ_d are predicted by the network, and variances σ^2 are all the same. Write an expression for the likelihood $P(\mathbf{y}|\mathbf{f}[\mathbf{x}, \boldsymbol{\phi}])$ for this model. Show that minimizing the negative log-likelihood of this model is still equivalent to minimizing a sum of squared terms if we don't estimate the variance σ^2.

Given this setup, the likelihood of observing a particular $\mathbf{y}$ given model predictions $\mathbf{f}[\mathbf{x}, \boldsymbol{\phi}]$, where ϕ are the parameters of the model, can be expressed as:

$$P(\mathbf{y}|\mathbf{f}[\mathbf{x}, \boldsymbol{\phi}]) = \prod_{d=1}^{10}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[\frac{-(y_d-\mu_d)^2}{2\sigma^2}\right]$$

The NLL is therefore:

$$\begin{aligned}
-\log P(\mathbf{y}|\mathbf{f}[\mathbf{x},\boldsymbol{\phi}]) &= -\log\left(\prod_{d=1}^{10}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[\frac{-(y_d-\mu_d)^2}{2\sigma^2}\right]\right)\\
&= -\sum_{d=1}^{10}\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[\frac{-(y_d-\mu_d)^2}{2\sigma^2}\right]\right)\\
&= \sum_{d=1}^{10}\frac{(y_d-\mu_d)^2}{2\sigma^2}\\
&= \sum_{d=1}^{10}(y_d-\mu_d)^2
\end{aligned}$$

which is equivalent to minimising the sum of squared term (if we don't estimate the variance σ^2).

Problem 5.8

Construct a loss function for making multivariate prediction $\mathbf{y}$ based on independent normal distributions with different variances σ_d^2 for each dimension. Assume a heteroscedastic model so that both means μ_d and variances σ_d^2 vary as a function of the data

$$L = -\sum_{i=1}^{I}\log\left(\prod_{d=1}^{D_i}\frac{1}{\sqrt{2\pi\mathbf{f}_{2d}[\mathbf{x}_i,\boldsymbol{\phi}]^2}}\exp\left[\frac{-(y_{id}-\mathbf{f}_{1d}[\mathbf{x}_i,\boldsymbol{\phi}])^2}{2\mathbf{f}_{2d}[\mathbf{x}_i,\boldsymbol{\phi}]^2}\right]\right)$$

Problem 5.9

Consider a multivariate regression problem in which we predict the height of a person in meters and their weight in kilos from data $\mathbf{x}$. Here, the units take quite different ranges. What problems do you see this causing? Propose two solutions to these problems.

The height uses different units than the weight and the numbers will be smaller. Consequently, the least squares loss will focus much more on the weight than the height. Two possible solutions are (i) to rescale the outputs so that they have the same standard deviation, build a model that predicts the rescaled outputs, and scale them back after inference or (ii) learn a separate variance for the two dimensions so that the model can automatically take care of this. The second approach can be done in either a homoscedastic or heteroscedastic context.

Problem 5.10

Extend the problem from 5.3 to predict both the wind direction and the wind speed and define the associated loss function.

$$L = \sum_{i=1}^{I} -\log\left[\frac{\exp[\mathbf{f}_2[\mathbf{x}_i, \phi]\cos(y_i - \mathbf{f}_1[\mathbf{x}_i, \phi]]}{2\pi \cdot \text{Bessel}_0[\mathbf{f}_2[\mathbf{x}_i, \phi]]}\right]$$