1 Overview

First, we’ll do a quick review of multiclass classification. Then, noting that we have moved into the realm of optimizing over matrices, we’ll introduce the concept of matrix norms.

2 Multiclass Review

In the setting of multiclass logistic regression, recall that we have a sample set $S$, which consists of $m$ samples $x_i \in \mathbb{R}^d$ and labels $y_i \in [1, \ldots, k]$, and we’d like to learn a matrix $W \in \mathbb{R}^{k \times d}$, whose row are the vectors $w_1, \ldots, w_k \in \mathbb{R}^d$, which predicts label probabilities

$$
\Pr[\text{label } y | x] = \frac{\exp(w_y^T x)}{\sum_{j=1}^k \exp(w_j^T x)}.
$$

Each $W$ specifies a model, and we’d like to find the one that maximizes likelihood of the data, which is just a product of this expression for each data point:

$$
\Pr[S | W] = \prod_{i=1}^m \Pr[y_i | x_i]
$$

It is convenient to consider maximizing the logarithm of this quantity, since the product becomes a sum:

$$
\log \Pr[S | W] = \sum_{i=1}^m \log \Pr[y_i | x_i]
$$
And since we’d like to be consistent with the framework of convex optimization, we’ll use the negative log-likelihood as our loss function:

\[ \ell(W) = -\log \Pr[S | W] = - \sum_{i=1}^{m} \log \left( \frac{\exp(w^\top y_i x_i)}{\sum_{j=1}^{k} \exp(w^\top j x_i)} \right) \]

\[ = \sum_{i=1}^{m} \log \sum_{j=1}^{k} \exp(w^\top j x_i) - w^\top y_i x_i . \]

This is a convex function that takes in a matrix \( W^{k \times d} \), and returns a real number. In order to run (stochastic) gradient descent, we need to compute the gradient, which is a matrix. This is not that scary. Of course, it will suffice to compute a single term \( \nabla \ell_i(W) \).

This is not that scary: it’s essentially the same derivation as in the case of two-class logistic regression. Let’s do it step-by-step:

First, consider the derivative of the log-sum-exp function:

\[ \frac{d}{dx} \log(e^x + C) = \frac{e^x}{e^x + C} . \]

From this, we have

\[ \frac{\partial}{\partial u_i} \log \left( \sum_{j=1}^{k} e^{u_j} \right) = \frac{\partial}{\partial u_i} \log \left( e^{u_i} + \sum_{j \neq i} e^{u_j} \right) = \frac{e^{u_i}}{e^{u_i} + \sum_{j \neq i} e^{u_j}} = \frac{e^{u_i}}{\sum_{j=1}^{k} e^{u_j}} . \]

Thus, we have the gradient of the function \( L(u) : \mathbb{R}^k \rightarrow \mathbb{R} \) for which \( \ell_i(W) = L(W x_i) - 1^\top y_i W x_i \). Specifically,

\[ L(u) = \log \left( 1^\top \exp(u) \right) , \]

and

\[ \nabla L(u) = \frac{1}{1^\top \exp(u)} \cdot \exp(u) , \]

where \( 1 \) is the all-ones vector, and \( \exp(\cdot) \) denotes the entrywise exponential.

Now, we are almost done. Compute the partial derivative with respect to each entry of \( W \):

\[ \frac{\partial}{\partial W_{j,c}} \ell_i(W) = \frac{\partial}{\partial W_{j,c}} \left( L(W x_i) - [W x_i]_{y_i} \right) \]

\[ = \left( \nabla L(W x_i) \right)_{j} - 1_{y_i=j} \cdot [x_i]_c . \]

Here, the indicator \( 1_{y_i=j} \) is 1 when we are computing a partial derivative in row \( y_i \), and 0 otherwise.

We have sneakily proven the matrix chain rule in general, which gives us the gradient in a more concise form:

\[ \nabla [W \mapsto L(W x_i)] = \nabla L(W x_i) x_i^\top . \]

From this, we can apply GD (sum each \( \nabla \ell_i \) at each iteration), or SGD (pick one).
3 Matrix Norms

When we considered binary classification or single-output regression, the parameters we optimized over always took the form of a vector $w$. In class, we considered imposing a norm constraint on this $w$; recall that we could solve this constrained optimization problem using projected gradient descent.

3.1 Norms

So far, we’ve encountered the $\ell_p$ norm of a vector:

$$\|x\|_p = \left( \sum_{i=1}^{d} |x_i|^p \right)^{1/p},$$

which generalizes some simple notions of magnitude of a vector: Euclidean ($p = 2$), Manhattan ($p = 1$), and largest-magnitude entry ($p \to \infty$).

In general, a norm $\| \cdot \|$ is a function from a real vector space $V$ to the non-negative reals $\mathbb{R}^+$ with the following properties:

- The zero vector has norm 0, and all others have positive norm.
- Homogeneity: $\|cx\| = |c| \cdot \|x\|$, $\forall x \in V$.
- Triangle inequality: $\|x + y\| \leq \|x\| + \|y\|$, $\forall x, y \in V$.

Note that by these properties, $\|x\|$ is always a convex function, and thus, the sublevel sets $\{ x : \|x\| \leq C \}$ are also convex. So, given a convex optimization problem we know how to solve, we can add a constraint or regularization by any norm we like (as long as we can compute projections or gradients, respectively).

Some examples of norms:

- $\|x\| = 7\|x\|_1 + 42\|x\|_\infty$. In general, positive linear combinations of norms are norms.
- $\|x\| = \sqrt{x_1^2 + 2x_2^2 + 3x_3^2 + \ldots + dx_d^2}$; In general, any $x \mapsto \|Ax\|$ is a norm, where $A$ is an invertible matrix.

3.2 Matrix norms

This leads us to a natural question: what natural norms exist for a matrix $M \in \mathbb{R}^{m \times n}$?

A silly-sounding but valid answer: treat $M$ like an $mn$-dimensional vector; then any vector norm of $M$ works. The $\ell_2$ version has a special name: the Frobenius norm, defined by

$$\|M\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} M_{ij}^2}.$$
Here, the $\ell_\infty$ case is just the largest absolute value of an entry in the entire matrix. In these cases, projection onto the constraint set $\|M\| \leq C$ is just as easy as vector projection.

However, the interpretation of vectorizing a matrix is sometimes unclear, especially when the matrix in question describes a linear map (say, in regression).

### 3.3 The operator norm

A more natural class of norms is the operator norm. Intuitively, viewing $M$ as a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$, an operator norm asks: “what’s the largest factor by which $M$ can blow up the magnitude of a vector?” In a formula (letting all vector norms be Euclidean): 

$$\|M\|_{\text{op}} := \sup_{v \in \mathbb{R}^n} \frac{\|Mv\|}{\|v\|} = \sup_{\|v\|=1} \|Mv\|.$$ 

A nice property is that the operator norm is submultiplicative: that is, $\|AB\|_{\text{op}} \leq \|A\|_{\text{op}} \cdot \|B\|_{\text{op}}$ under matrix multiplication. Recall that a norm is only required to be subadditive.

It’s a little less clear how to compute this norm. Thankfully, we don’t have to do a brute-force search over all test vectors $v$. Recalling the Rayleigh quotient (sometimes known as variational) characterization of eigenvalues:

$$\|M\|_{\text{op}}^2 = \sup_{v \in \mathbb{R}^n} \frac{v^\top M^\top Mv}{v^\top v} = \lambda_{\text{max}}(M^\top M).$$

So, we can measure this norm by a maximum-eigenvalue computation. This is why the operator norm is sometimes also known as the spectral norm.

Unfortunately, projection and gradient are now more complicated matters. However, it can be verified that the Frobenius norm is always an upper bound for the operator norm. So, this allows us to say that a Frobenius norm constraint also acts as an operator norm constraint; the latter is often more interpretable.

### 3.4 Subordinate norms: generalizing operator norms

In defining the operator norm, we sneakily made two arbitrary choices: that we measured the “blowup” in terms of the Euclidean norms of both $v$ and $Mv$. Indeed, we can obtain a whole family of operator norms, by varying the way we measure the size of each vector in defining the blowup factor. We usually consider different $\ell_p$ norms:

$$\|M\|_{p \to q} := \sup_{v \in \mathbb{R}^n} \frac{\|Mv\|_q}{\|v\|_p} = \sup_{\|v\|_p=1} \|Mv\|_q.$$ 

So, the original definition of operator norm is recovered by setting $p = q = 2$.

Let’s see what happens when we set $p = 1, q = 2$. Then, take a moment to convince yourself that $\|Mv\|_{1 \to 2}$ is simply the largest $\ell_2$ norm of any column of $M$. Such a constraint is easier to check and enforce than the operator norm, and the gradient of this quantity is easy to compute.

\[ \text{One-line proof: } \|M\|_F = \text{tr}(M^\top M) = \sum \lambda_i(M^\top M) \geq \max \lambda_i(M^\top M) = \|M\|_{\text{op}}^2. \]