1 Topological Data Analysis

1.1 Theoretical Background

1.1.1 Simplices

A collection of k + 1 points is affinely independent if the members lie on an affine hyperplane of dimension k. The convex hull of a set $X \subseteq \mathbb{R}^n$ is the smallest convex set containing X. For a finite set $\{x_0, \ldots, x_k\}$ this is the set of combinations $\sum_{i=0}^{k} \lambda_i x_k$ with $\sum_{i=0}^{k} \lambda_i = 1$.

Definition 1.1. Given an affinely independent set $X = \{x_0, \ldots, x_k\} \subseteq \mathbb{R}^n$, the k-dimensional simplex (sometimes called a geometric simplex) $\sigma = [x_0, \ldots, x_n]$ spanned by X is the convex hull of X. The points of X are called the vertices of σ, and the simplices spanned by subsets of X (which are necessarily affinely independent) are called the faces of σ.

Definition 1.2. A simplicial complex K is a finite collection of geometric simplices such that

(i) for any simplex $\sigma \in K$, every face of σ is in K;

(ii) for any two simplices $\sigma, \tau \in K$, $\sigma \cap \tau$ is either empty, or a face of both σ and τ.

The dimension of K is the largest dimension of any simplex in K. A subcomplex of K is a subset of K which is a simplicial complex.

1.1.2 Simplicial Homology

Definition 1.3. Given a simplicial complex K, we define a p-chain as a subset of the p-simplices in K.

We can write a p-chain c as the formal sum $c = \sum a_i \sigma_i$ where the sum is over all p-simplices and the coefficients are in $\mathbb{Z}_2$. This gives rise to an abelian group $(C_p, +)$, where $c + c' = \sum (a_i + b_i)\sigma_i$ and the coefficients are reduced mod 2. It can be further extended to a vector space by defining scalar multiplication as $a \cdot c = \sum (a \cdot a_i)\sigma_i$.
The boundary of a p-simplex is the set of (p-1)-faces. The boundary of a p-chain is the sum of the boundaries of its p-simplices: $\partial_p c = \sum a_i \partial_p \sigma_i$.

Definition 1.4. This can be formalised as an operation between vector spaces:

$$\partial_p : C_p \to C_{p-1}$$

called the boundary homomorphism.

These vector spaces and maps can be lined up into a sequence

$$\cdots \to \partial_{p+2} C_{p+1} \to \partial_{p+1} C_p \to \partial_p C_{p-1} \to \partial_{p-1} \ldots$$

called the chain complex of K.

Proposition 1.1. $\partial \partial c = 0$.

Proof. Let σ be a p-simplex. The vertices of a (p-2)-face are a subset of size p-1 from the p+1 vertices of σ. There are two subsets of $V(p)$ with size p containing these p-2 vertices. It follows that every (p-2)-face of σ is contained in exactly two (p-1)-faces, and therefore that $\partial\partial$ vanishes on σ as coefficients are reduced mod 2. By homomorphism properties this now follows for p-chains. □

Definition 1.5. A p-cycle is a p-chain without boundary, the set of all p-cycles is therefore $\ker \partial_p$. As the kernel of a homomorphism it is a subgroup of C_p, and we denote it by Z_p.
Similarly, we define a p-boundary to be the boundary of a (p+1)-chain, the set of all p-boundaries is therefore $\mathrm{im} \partial_{p+1}$. As the image of a homomorphism it is a subgroup of C_p, and we denote it by B_p.

Note also that $B_p \trianglelefteq Z_p$ because Z_p is abelian, we can therefore take the quotient Z_p / B_p which represents the distinct cycles up to boundary.

Definition 1.6. We say that $z, z' \in Z_p$ are homologous if they fall in the same conjugacy class in Z_p / B_p.

We denote the quotient Z_p/B_p by H_p, and call it the p^{th} homology group. Its members are referred to as homology classes.
The above definitions can be extended to incorporate the vector space structure. Then, the rank-nullity theorem can then be applied to give

$$\mathrm{rank}H_p = \mathrm{rank}Z_p - \mathrm{rank}B_p,$$

which we refer to the p^{th} Betti number of K and notate by $\beta_p = rankH_p$.

Definition 1.7. Define the Euler characteristic of a simplicial complex K by

$$\chi(K) = \sum_{i=0}^{k} (-1)^i \mathrm{rank}C_p,$$

where $k = \mathrm{dim}K$.

The map $\partial_p : C_p \to B_{p-1}$ is surjective, so we can apply the rank-nullity theorem to get $\mathrm{rank}B_{p-1} = \mathrm{rank}C_p - \mathrm{rank}Z_p$. We can use this, along with that $B_i = \mathbf{0}$ for i outside $\{0, \ldots, k-1\}$, to rewrite the Euler characteristic:

$$\begin{aligned}
\chi(K) &= \sum_{i=0}^{k} (-1)^i (\mathrm{rank}Z_p + \mathrm{rank}B_{i-1}) \\
&= \sum_{i=0}^{k} (-1)^i \mathrm{rank}Z_p - \sum_{i=0}^{k} (-1)^i B_i \\
&= \sum_{i=0}^{k} (-1)^i (\mathrm{rank}Z_p - B_i) \\
&= \sum_{i=0}^{k} (-1)^i \beta_i.
\end{aligned}$$

1.1.3 Homology

Definition 1.8. The underlying space of a simplicial complex K is defined as the union

$$|K| = \bigcup_{\sigma \in K} \sigma,$$

and equipt with the subspace topology.

Definition 1.9. A triangulation of a topological space X is a simplicial complex K, whose underlying space is homeomorphic to X.

The previous definition of the Euler characteristic can be extended to be a well-defined topological invariant by defining it on a triangulation of a space. It is independent of the specific choice of triangulation. We note that having the same homology groups is weaker than having the same homotopy type, which is again weaker than being homeomorphic:

$$X \approx Y \implies X \simeq Y \implies H_p(X) \cong H_p(Y) \text{ for all p..}$$

The implications of this are that to compute the Betti numbers of X, we may find a space Y with the same homotopy type and compute its Betti numbers.

1.2 Complex Construction

1.2.1 Abstract Simplicial Complexes

Definition 1.10. An abstract simplicial complex A is a system of sets such that $\alpha \in A$ and $\beta \subseteq \alpha$ implies $\beta \in A$.

The sets α are called abstract simplices, their dimension is defined as their cardinality minus one. The dimension of an abstract simplex A is defined as the dimension of its largest member.

Definition 1.11. A geometric realisation of an abstract simplicial complex A is an embedding of A in Euclidean space as a simplical complex. By this we mean that each vertex in A is associated with a distinct point in Euclidean space, with the property that the system of subcollections of these points corresponding to A is a simplicial complex.

It is natural to ask whether every abstract simplical complex has a geometric realisation in $\mathbb{R}^d$. The following result is in the affirmative if d is big enough.

Theorem 1.1. *Any abstract simplicial complex of dimension k has a geometric realisation in $\mathbb{R}^{2k+1}$.*

1.2.2 Nerves

Definition 1.12. Let X be a finite collection of sets. The nerve of X is the system of subcollections of X whose sets have a non-empty common intersection:

$$\mathrm{Nrv}X = \{V \subseteq X : V \neq \emptyset \text{ and } \textstyle\bigcap_{v\in V} v \neq \emptyset\}.$$

A nerve is an abstract simplicial complex because if $V \in \mathrm{Nrv}X$ and $W \subseteq V$, with W nonempty, then $\bigcap_{w\in W} w \supseteq \bigcap_{v\in V} v \supsetneq \emptyset$ so $W \in \mathrm{Nrv}X$.

Theorem 1.2. *If all sets in X are closed and triangulable, and all non-empty common intersections of the sets are contractible, then $NrvX$ and $\bigcup_{x\in X} x$ have the same homotopy type.*

1.2.3 Alpha Complexes

Let S be a finite set of points in $\mathbb{R}^2$. We refer to the members of S as sites to distinguish them from points in the surrounding space.
Fix $\alpha > 0$, and write $B_x(\alpha)$ for the closed ball with radius α centred at $x \in \mathbb{R}^2$. It is called empty if $D_x(\alpha) \cap S = \emptyset$.
A point x is the centre of an empty disc with a radius α if and only if it is further than α from every site. We consider the union of discs of radius α centred at the sites:

$$U_S(\alpha) = \bigcup_{s\in S} D_s(\alpha).$$

This union is the complement of the set of centres of the empty discs.

Definition 1.13. For $s \in S$, define the Voronoi region of s as

$$V_s = \{x \in \mathbb{R}^2 : \|x - s\| \leq \|x - t\|, \forall t \in S\}.$$

Notice that $\|x - s\| \leq \|x - t\|$ is a closed half-plane, so V_s is the intersection of finitely many half-planes and therefore a convex polygon. Any two Voronoi regions intersect at most along their boundaries, and together, the Voronoi regions cover the entire plane.

Definition 1.14. The Voronoi diagram of S is the set of Voronoi regions, one for each site in S.

Definition 1.15. Given a Voronoi diagram of $S \subseteq \mathbb{R}^2$ we can construct its Delaunay triangulation by connecting two sites with a straight edge whenever the corresponding two Voronoi regions share an edge:

$$\{T \subseteq S : T \neq \emptyset \text{ and } \textstyle\bigcap_{s\in T} V_s \neq \emptyset\}$$

Write $R_s(\alpha) = V_s \cap D_s(\alpha)$, noting that it is a convex set as the intersection of convex sets. Furthermore, $U_S(\alpha) = \bigcup_{s\in S} R_s(\alpha)$, so that the sets $R_s(\alpha)$ cover $U_s(\alpha)$. Furthermore, the common overlap of the regions is limited to shared edges and vertices.
Following the recipe for the Delaunay triangulation, we construct the α-complex by drawing an edge between two sites s and s' if the intersection of $R_s(\alpha)$ and $R_{s'}(\alpha)$ is a common edge. We denote this complex by $A_S(\alpha)$ or $A(\alpha)$.

Definition 1.16. To be more precise in the complex structure that $A(\alpha)$ has, we formally define it as the nerve of the Voronoi diagram of S:

$$A(\alpha) = \{\sigma \subseteq S : \textstyle\bigcap_{s\in\sigma} R_s(\alpha) \neq \emptyset\}.$$

Definition 1.17. The α-shape of S is defined as the union of all simplices in $A_S(\alpha)$.

Let X be the Voronoi diagram of S. The members of X, the sets $R_s(\alpha)$, are all closed (as finite intersections of closed sets) and triangulable (using the above triangulation). Then, applying the Nerve Theorem, $U_s(\alpha)$ and $A(\alpha)$ have the same homotopy type.

1.2.4 Čech Complexes

Suppose we simplify the construction of the α-complex by considering only the intersection of discs, without first restricting them to the corresponding Voronoi regions. The Čech complex formalises this idea.

Definition 1.18. Using the same notation as before, define the Čech complex as

$$\check{C}ech(r) = \{T \subseteq S : T \neq \emptyset \text{ and } \bigcap_{s\in T} D_s(r) \neq \emptyset\}.$$

The Čech complex is isomorphic to the nerve of the discs $\text{Nrv}(\{D_s(r) : s \in S\})$. By the Nerve Theorem, the Čech complex has the same homotopy type as the union of the discs (and therefore $|A(r)| \simeq |\check{\text{C}}\text{ech}(r)|$).

1.2.5 Vietoris-Rips Complexes

It can be difficult (or impossible in some metric spaces) to test whether a collection of disks have a non-empty intersection. We now define a complex that needs only the distances between points in S for its construction.

Definition 1.19. Using the same notation as before, define the Vietoris-Rips complex as the set of abstract simplexes σ with vertices S, such that any two vertices in σ are at most a distance of $2r$ from each other. We denote the complex by Vietoris-Rips(r)

The Vietoris-Rips complex doesn't have the same homotopy type as the union of disks of radius r, which suggests it can have topological artifacts that do not show up in the data. While this is true, in practice, these artifacts tend to be limited.

Proposition 1.2. Let S be a finite set in $\mathbb{R}^2$. Then $\check{\text{C}}\text{ech}(r) \subseteq \text{Vietoris-Rips}(r) \subseteq \check{\text{C}}\text{ech}(r')$, where $r' = \frac{2r}{\sqrt{3}}$.

Proof. For $\sigma \in \check{\text{C}}\text{ech}(r)$, $\bigcap_{s\in V(\sigma)} D_s(r) \neq \emptyset$, therefore, by the triangle inequality, the distance between any two points in $|\sigma|$ is at most $2r$, so $\sigma \in \text{Vietoris-Rips}(r)$.
Next, fix $\sigma \in \text{Vietoris-Rips}(r)$. The second inclusion is trivial if the dimension of σ is zero or one, so assume it's two. Furthermore, we can assume that the three points in $V(\sigma)$ are the vertices of an equilateral triangle with side lengths $2r$. The circumcenter of this triangle is a distance of r' from the vertices, therefore $\sigma \in \check{\text{C}}\text{ech}(r')$. □

1.3 Persistent Homology

1.3.1 Filtration

Definition 1.20. Let K be a simplicial complex. A filtration on K is a sequence of simplicial complexes:

$$\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K.$$

Example 1.1. Let K be the Delaunay triangulation of a finite set $S \subseteq \mathbb{R}^2$. For each simplex $\sigma \in K$ there is a real number α_σ such that σ belongs to $A(\alpha)$ if and only if $\alpha_\sigma \leq \alpha$. We can order the n simplices in K, first by dimension, then within each dimension group using the order induced by $\alpha_{\sigma_1} \leq \alpha_{\sigma_2} \leq \ldots \alpha_{\sigma_n}$. This gives a filtration on K:

$$\emptyset = K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n = K.$$

This filtration is called flat because any two contiguous complexes differ by a single simplex. Since every alpha complex is a subcomplex of the Delaunay triangulation, every alpha complex belongs to this triangulation. Moreover $\alpha \leq \alpha'$ implies $A(\alpha) \subseteq A(\alpha')$

1.3.2 Birth and Death

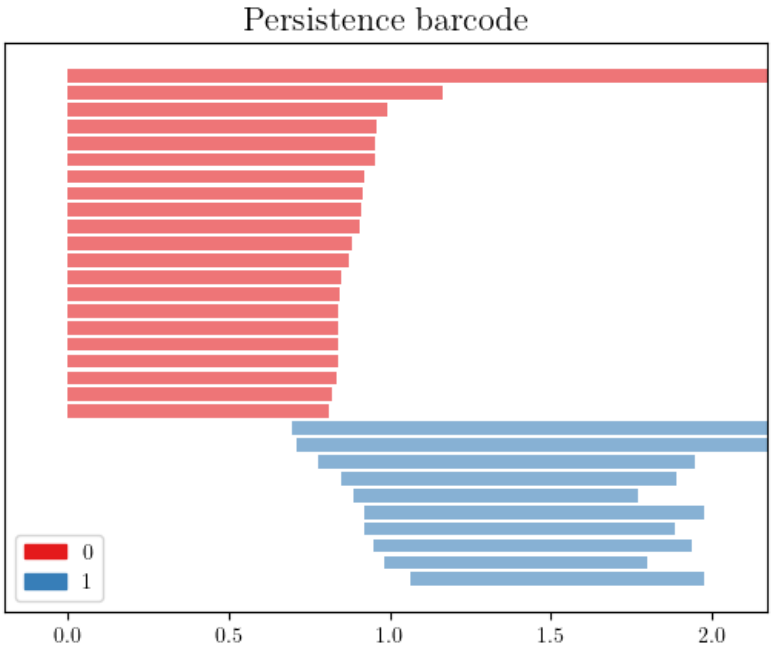
Let $K_0 \subseteq K_1 \subseteq \ldots \subseteq K_n$ be a filtration. For $i \leq j$ there is an inclusion $f^{i,j} : K_i \hookrightarrow K_j$. This map induces an inclusion homomorphism $f_p^{i,j} : Z_p(K_i) \hookrightarrow Z_p(K_j)$, which in turn induces the homomorphism

$$\phi_p^{i,j} : H_p(K_i) \to H_p(K_J).$$

The image of $\phi_p^{i,j}$ is called a persistent homology group. It contains all the p-dimensional homology classes from K_i that are still present in K_j.
We say a class $\gamma \in H_p(K_i)$ is born at K_i if it's not in the image of $\phi_p^{i-1,i}$, and a class born in K_i dies entering K_{j+1} if $\phi_p^{i,j}(\gamma)$ isn't in the image of $\phi_p^{i-1,j}$ but $\phi_p^{i,j+1}(\gamma)$ is in the image of $\phi_p^{i-1,j+1}$. The intuition behind the second part of this definition is that we characterise the death of a class as it being subsumed by a class which existed prior to its birth. The first part of this definition is to ensure this subsumption happens exactly at K_{j+1} and not before.
The index of persistence of γ is $j + 1 - i$. Often we have a function which governs the construction of the filtration. Let b be the value of the function at a classes birth and d the value at its death. We call (b, d) a birth death pair, and difference $d - b$ the persistence of the class. In an alpha complex filtration, this would be the extension of $\sigma \mapsto \alpha_\sigma$ to the filtration (which is possible because it is flat).

1.3.3 Barcodes

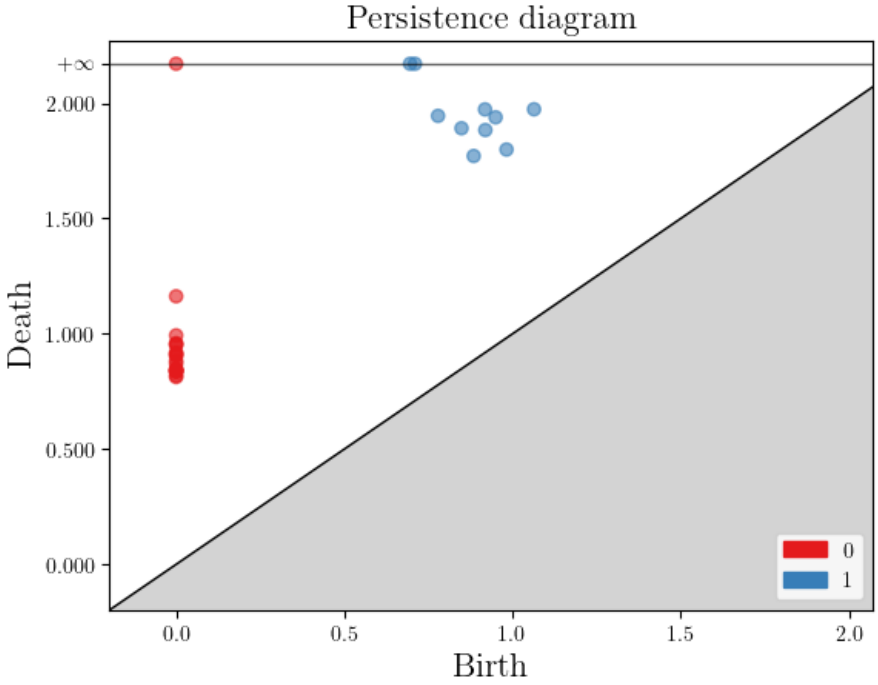
A barcode plot is a convenient way to represent the birth and death of classes throughout a filtration. In its unrestricted form, each horizonal bar represents a generator of the homology group, its length and position represent when, and for how long, it was alive. Bars which extend to the right hand edge of the diagram are those which persisted to the end of the filtration.
The number of bars tends to get large when the data has more complicated topology. We tend to simplify the diagram by placing restrictions on the minimum length of bars we'd like to include, or the total number of bars.
The following plot shows a simplified barcode plot for 1000 points sampled from a torus. Importantly, we can see that a single bar in the zeroth homology class extends to infinity - which represents the single connected component of a torus, and two bars in the first homology class that extend to infinity represent the generators of $Z \times Z = H_1(T^2)$.



1.3.4 Peristence Diagrams

As mentioned, barcode diagrams can become extremely complicated for data with more complex topology. For these cases, a representation as points in the plane is convenient. In a peristence diagram

we plot birth time against death time with each point representing a generator from a homology class. Points are colour coded by the homology group they belong to.
The following plot shows a persistence diagram for 1000 points sampled froma torus. The points close to the line $y = x$ represent generators which which died shortly after being born. Those far away from this line represent generators which persisted for longer. We can again see some of the important topological features of the torus encoded in the diagram.



We now define a metric on the space of persistence diagrams.

Definition 1.21. Let $\mathcal{D}$ and $\mathcal{D}'$ be persistence diagrams. Define the Bottleneck distance between them as

$$W_\infty(\mathcal{D}, \mathcal{D}') = \inf_{\mu:\mathcal{D}\to\mathcal{D}'} \max_{x\in\mathcal{D}} \|x - \mu(x)\|_\infty,$$

where μ is a bijection.

Definition 1.22. Similarly, we define the p-th Wasserstein distance as

$$W_p(\mathcal{D}, \mathcal{D}') = \inf_{\mu:\mathcal{D}\to\mathcal{D}'} \sum_{x\in X} \|x - \mu(x)\|_\infty^q.$$

Also, we can define the following norms of a persistence diagram:

Definition 1.23.

$$\|\mathcal{D}\|_\infty = \max_{x,y\in\mathcal{D}} \|x - y\|,$$

Definition 1.24.

$$\|\mathcal{D}\|_p = \sqrt[p]{\sum_{x,y\in\mathcal{D}} \|x - y\|^p}.$$

1.3.5 Peristence Landscapes

Fix a filtration, and consider the birth death pairs $\{(b_i, d_i)\}_{i=1}^n$ for a fixed homology dimension p. For each birth and death pair (b, d), define the piecewise linear function $f_{(b,d)} : \mathbb{R} \to \infty$ by

$$f_{(b,d)} = \begin{cases} 0 & x \notin (b, d) \\ x - b & x \in (b, \frac{b+d}{2}) \\ d - x & x \in (\frac{b+d}{2}, d). \end{cases}$$

So $f_{(b,d)}$ is a steeple function, with the steeple on (b, d), and the height of the steeple proportional to $d - b$

Define the pth persistence landscape of the data as $L : N \times \mathbb{R} \to [0, \infty)$ by setting $L(k, x)$ as the kth largest of the values $\{f_{(b_i,d_i)}(x)\}_{i=1}^{n}$, and 0 if k exceeds n.

2 Machine Learning

2.1 Clustering

2.2 Regression

2.3 Clustering

2.4 Dimensionality Reduction

3 Deep Learning

3.1 Supervised Learning

The setting of supervised learning refers to the prediction of an output y from an input x. Consider a set of N training examples $\{(x_1, y_1), \ldots, (x_N, y_N)\}$. A learning algorithm seeks a function $h : X \to Y$ which provides a good model for the data: $h(x) \approx y$.

3.1.1 Loss Functions

Suppose we restrict our attention to a parameterised family of functions $h_\theta : X \to Y$. We can define a loss function J_θ which quantifies how good h_θ is as a model of our data, with low values indicating a better model.

Example 3.1. Suppose that y is binary, and that $t = h_\theta(x)$ represents a probability that the label for x is one. We can define

$$\mathcal{J}_\theta^i = -y_i \log t_i - (1 - y_i) \log(1 - t_i),$$

which penalizes "confident" incorrect predictions (i.e. t close to $1 - y$) more than "low confidence" incorrect predictions (i.e. t close to $\frac{1}{2}$). The cross-entropy loss is then defined as

$$\mathcal{J}_\theta = \frac{1}{N} \sum_{i=1}^{N} \mathcal{J}_\theta^i.$$

Example 3.2. The previous example can be extended to the case when y is categorical. Suppose that there are k classes which y can belong to, so that $y \in \{0, 1\}^k$ with only one nonzero entry indicating the class y belongs to. Suppose also that t^j $(1 \leq j \leq k)$ represents a probability that the the label of x is j, so then $\sum_{j=1}^{k} t^j = 1$. Then we can define a multidimensional analog of $\mathcal{J}_\theta^i$

$$\mathcal{J}_\theta^i = -\sum_{j=1}^{k} y_i^j \log t_i^j.$$

Then we can define the cross-entropy loss accross the training examples as

$$\mathcal{J}_\theta = \frac{1}{N} \sum_{i=1}^{N} \mathcal{J}_\theta^i$$

Example 3.3. Suppose that y is now numerical, specifically $y \in \mathbb{R}^k$. Suppose also that the range of h_θ is $\mathbb{R}^k$. Write

$$\mathcal{J}_\theta^i = \|y_i - t_i\|^2.$$

Then we can define the mean squared error as

$$\mathcal{J}_\theta = \frac{1}{N} \sum_{i=1}^{N} \mathcal{J}_\theta^i.$$

3.1.2 Optimisation

Since a low value of the loss function J_θ indicates that h_θ is a better model of the data, we aim to minimise the loss function. In simple cases, like for a linear regression with mean squared error loss, we can find a closed form expression for the global minima. In more complicated cases this is not possible, so use a $\mathcal{J}_\theta$ is differentiable in θ and apply an optimisation algorithm to find a minima.

Example 3.4. Recall from calculus that $\Delta\mathcal{J}_\theta$ points in the direction of steepest ascent. By linearity of the tangent space $-\Delta\mathcal{J}_\theta$ points in the direction of the steepest descent.
Then to find a minima of $\mathcal{J}_\theta$ we can continuously update θ with the rule

$$\theta - = \alpha \mathcal{J}_\theta,$$

where $\alpha > 0$ is refered to as the learning rate. The algorithm derived from this update rule is called gradient descent.

Example 3.5. A variant of gradient descent which is less compute intensive is stochastic gradient descent. In stochastic gradient descent we sample i uniformly from $\{1, \ldots, N\}$ and update θ by the rule

$$\theta - = \alpha \Delta \mathcal{J}_\theta^i.$$

3.2 Neural Networks

Neural networks refer to non-linear models $h_\theta(x)$ that involve combinations of matrix multiplication and entry-wise non-linear operations. In this section we outline some common operations used in these models.

3.2.1 Layers

Example 3.6. Input Layer: The input layer receives the raw input data and passes it to the subsequent layers. It doesn't perform any computations and is typically designed to match the dimensionality of the input data.

Example 3.7. Fully Connected Layer: A fully connected layer connects every neuron in the current layer to every neuron in the subsequent layer. Each neuron in a connected layer receives inputs from all the neurons in the previous layer and performs a weighted sum, followed by an activation function.

Example 3.8. Convolutional Layer: Convolutional layers are commonly used in convolutional neural networks (CNNs) for processing grid-like input data, such as images. These layers apply convolution operations to the input data using learnable filters, capturing local patterns and spatial relationships.

Example 3.9. Pooling Layer: Pooling layers are often used in conjunction with convolutional layers in CNNs. They downsample the feature maps by aggregating nearby values, reducing spatial dimensions and extracting dominant features. Common pooling operations include max pooling and average pooling.

Example 3.10. Recurrent Layer: Recurrent layers, such as Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU), are used for sequential data processing. These layers have recurrent connections, allowing them to maintain and propagate information across time steps.

Example 3.11. Dropout Layer: Dropout layers help prevent overfitting in neural networks. They randomly set a fraction of the inputs to zero during training, forcing the network to learn more robust representations by not relying heavily on any particular set of input features.

Example 3.12. Batch Normalization Layer: Batch normalization layers normalize the activations of the previous layer, typically by subtracting the batch mean and dividing by the batch standard deviation. This helps stabilize and speed up the training process, making the network more robust to changes in input distributions.

Example 3.13. Activation Layer: Activation layers introduce non-linearities into the network by applying a non-linear activation function to the outputs of the previous layer. Common activation functions include ReLU (Rectified Linear Unit), sigmoid, and tanh.

Example 3.14. Output Layer: The output layer provides the final predictions or outputs of the neural network. The configuration of this layer depends on the type of task the network is designed for. For example, for classification tasks, a softmax layer is commonly used to produce probability distributions over classes.

3.3 Activation Functions

Example 3.15. Sigmoid Activation Function: The sigmoid activation function is defined as:

$$f(x) = \frac{1}{1 + e^{-x}}.$$

Example 3.16. Hyperbolic Tangent (Tanh) Activation Function: The Tanh activation function is defined as:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}.$$

Example 3.17. Rectified Linear Unit (ReLU) Activation Function: The ReLU activation function is defined as:

$$f(x) = \max(0, x).$$

Example 3.18. Leaky ReLU Activation Function: The Leaky ReLU activation function is defined as:

$$f(x) = \begin{cases} x, & \text{if } x \geq 0 \\ \alpha x, & \text{if } x < 0 \end{cases},$$

where α is a small constant, typically 0.01.

Example 3.19. Softmax Activation Function: The softmax activation function is defined as:

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}},$$

for $i = 1, 2, \ldots, N$, where N is the number of classes.

3.4 Common Architectures