Deep Learning

Computing with the metaphorical brain.

Nicholas Gale and Stephen Eglen

Deep Learning.

- Deep Learning typically refers to statistical models with many layers.
- The layers are composed of computational units called neurons.
- The architecture of the models is inspired by neural architectures.
- With many parameters and training examples models can achieve super human peformance in *some* tasks.

The hype

None of us today know how to get computers to learn with the speed and flexibility of a child. Andrew Ng, Deep Learning Pioneer

- The field can appear extremely fast: “ground-breaking” or “state-of-the-art (SOTA/SOA)” are released all the time.
- Be wary of this; it often amounts to adding more compute, parameter tweaking on a reduced dataset, or overfitting.

The response

Torch.manual_seed(3407) is all you need: On the influence of random seeds in deep learning architectures for computer vision

- There are often popular satirical responses debunking the hype.
- That aside: the field **does** move incredibly quickly and there **are** amazing and groundbreaking achievements routinely posted.
- Above all the models are *useful* and therefore worth examining.

Where are we headed?

```julia
using JLD2, Flux, Images, BSON
BSON.@load "./models/sharks/conv.bson" convolutional
data = JLD2.load("./models/sharks/data.jld")
pred = Flux.onecold(convolutional(data["batched"]), unique(data
display(hcat(pred[1:5], data["labels"][1:5]))
mosaicview(data["imgs"]...; nrow=1)
```

```
5×2 Matrix{Any}:
 "nurse"      "nurse"
 "thresher"   "thresher"
 "nurse"      "nurse"
 "thresher"   "thresher"
 "basking"    "basking"
```

Course Outline

1. Mathematical and statistical modelling.
2. Brain modelling.
3. Simple neural networks and what they mean.
4. Developing primitive networks from scratch.
5. Developer tools: Flux; PyTorch; Tensorflow.
6. Developing complex deep learning models.

Data

- Data is typically anything we can measure.
- It often is categorised by a set of real numbers (1.2, -1.4) and units (meters/second, red).
- The quantisation of data is useful because it allows us to perform formal mathematical operations on it.

Probability

- Probability is a number we use to characterise the likelihood of an event.
- The probability can be thought of as the proportion of times we expect this event to happen asymptotically. (Debate)
- The probability of standard a die rolling 1 is $1/6$.

Distribution

- A distribution is the probabilitisc description of *all* possible events.
- Distributions may be discrete (categorical) or continuous.
- All elements in the distribution must integrate to a total probability of 1.
- A distribution is often parameterised by a series of numbers: $\vec{\beta}$

PDFs and CDFs

- Distributions can be described with a probability density function (PDF) or cummulative density function (CDF):

$$CDF(x) = \int_{\infty}^{x} PDF(y) dy.$$

- We can sample from a distribution using the inverse of the CDF.
- We tend to describe random variables in terms of their distributions:

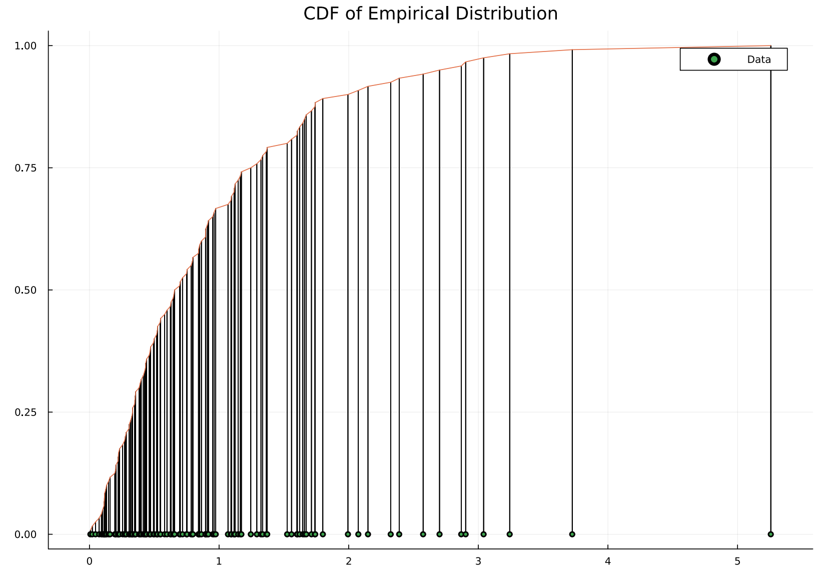
$$X \sim D(\vec{\beta})$$

Data as a Distribution

- We can think of data as being described by a distribution.
- A distribution is therefore a data generating process.
- Each data point (an image, a measurement of velocity) is a random sample from some (potentially unknown distribution)

Empirical Distributions

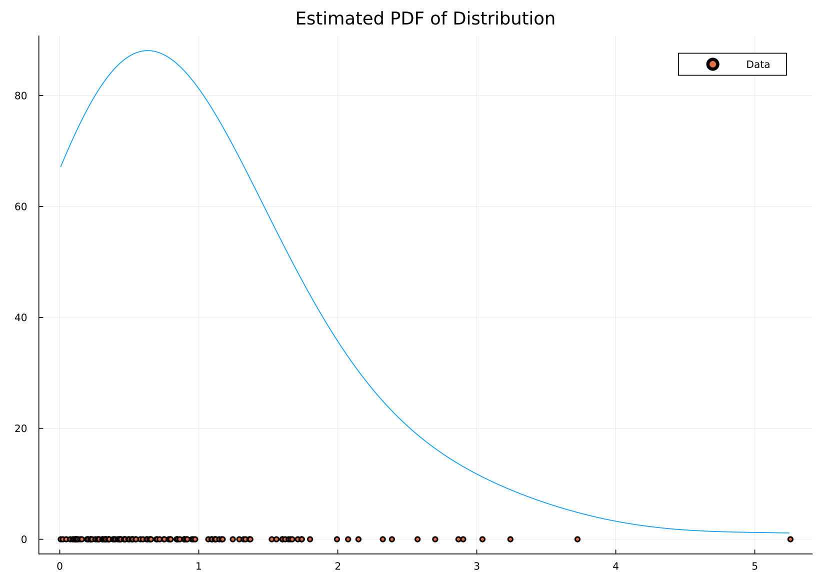
- We can use data samples to inform us about distributions.
- The most naive thing to say is that the data *is* the distribution.
- The distribution is then defined by N data points and the CDF increases by $1/n$ for each data point.



CDF of Empirical Distribution
1.00
0.75
0.50
0.25
0.00
0
1
2
3
4
5
Data

Kernel Density Estimation

- We might also say that each datum represents a kernel probability function.
- The kernel encodes the likelihood of sampling at that point e.g. $N(x, 1/n)$.
- The distribution may then be estimated by summing the kernels and normalising by the number of data.



Estimated PDF of Distribution
Data
0
20
40
60
80
0
1
2
3
4
5

Statistics

- Data distributions can also be characterised by statistics.
- Statistics are measurements that can be made on a data set.
- They are also known (or can be estimated) for distributions.
- The average, variance, standard deviation, quartiles, mode, and median are common examples.
- A known distribution can be estimated using statistics: e.g $N_{\text{est}}(\mu = \text{av}, \sigma^2 = \text{var})$

Central Limit Theorem

- Central limit theorem: the mean of a collection of indepedent measurements tends towards a normal distribution.
- Let $\{X_i\}_{i=1}^{n}$ be iid from a distribution with mean μ and variance σ^2.
- Suppose $Z = \frac{\sum_i \frac{X_i}{n} - \mu}{\frac{\sigma}{\sqrt{n}}}$. Then, $Z \sim N(0, 1)$.
- This is useful for analysing statistics (estimating distributions, regression, etc.)

Modelling the Real World

- A model is a reduced description of real world phenenoma. They are *always* wrong, but sometimes useful.
- They are used to *explain* and *predict* aspects of that phenenoma.
- They can be words, pictures, mental, mathematical, or algorithmic/computational.
- We like mathematical and computational models because they are *precise* i.e. no ambiguity and falsifiable by experiment.

Mathematical and Computational Models

- The general form of a precise model is a functional relationship.
- The model (f) takes input (x) and as a black box produces output (y): $y = f(x)$
- The science (or description) is encoded in the definition of x.

Statistical Models

- We would like to relate our measured data to each other.
- We do this by asserting that there is a model between the relevant data.
- Then, we precisely formulate this model in the form of a mathematical function.
- We typically manipulate an *indepedent* variable (x) and measure the *depedent* variable (y).
- We *assume* with some random error ϵ:

$$y_i = f(x_i) + \epsilon$$

Transformed Distribution

- We can also query the statistics of the transformed variable (y = f(x)) under the model:

$$F_Y(y(x)) = F_X(x)$$

- Differentiating CDFs gives PDFs:

$$\frac{\partial}{\partial x} F_Y(y(x)) = f_X(x)$$

- Using the chain rule yields:

$$f_Y(y) = \left|\frac{\partial y}{\partial x}\right|^{-1} f_X(x)$$

Models are Data Generating

- We can think of a model as a data generating process.
- A measurement is made with some error: it is a sample from a distribution.
- Data is a linearly indepedent collection of measurements. (Central Limit Theorem)
- Model makes predictions by transforming the indepedent measurements. Depedent measurements come from the *true* process.
- Models *often* (not necessarily) take some natural law form of this process e.g. $x(t) = x_0 + vt$

Parameters and Hyperparameters

- A model is characterised by a series of numbers that are not data.
- These are typically called *parameters* e.g. v, x_0 in $x(t) = x_0 + vt$.
- A models parameters may come from a distribution and these are referred to as *hyper-parameters*
- Hyper-parmaters can also refer to parameterisations that are part of the model specification e.g. fitting.

Estimating Parameters

- We generally use data to try and estimate the parameters of a model.
- This is a procedure known as *fitting*.
- Fitting typically involves minimising a quantity between predictions and measurements.

Bayesian Statistics vs Frequentists

- Frequentist statistics assume estimates are true in the limiting value of large data.
- Bayesian statistics assume that an estimate has a probability and data updates the probability via Bayes rule:

$$p(\alpha|\text{data}) = \frac{p(\text{data}|\alpha)p(\alpha)}{\text{data}}$$

  .

- $p(\alpha|\text{data})$ is the *posterior* probability model after observration
- $p(\alpha)$ encodes the *prior* belief that a parameter α follows a given distribution.
- Bayesian statistics allow us to encode uncertainty into our data and treat our data and parameters as distributions.

Model Fits

- A model relates the distribution of the indepedent/depedent variables.
- We also sample these distributions in the form of data.
- We generally want to minimise some quantity error quantity between these two.
- For example this could be minimising least squared error, or likelihood maximisation under a given model.

Goodness of fit

- To assess the goodness of fit we usually look at the difference between these distributions.
- Most often this is done through the covariance e.g.

$$r^2_{\text{Pearsons}} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}$$

- Be careful with goodness of fit. Pearsons for example will only be meaningful for linear regression.
- Many other measures, some penalise number of parameters e.g. Akaike Information Criterion.

Linear Regression

- Linear regression is the most common form of model fitting.
- It assumes that the *parameters* are linear in the model.
- The regressors (depedent variables) may still be non linear e.g. $y = \beta_0 + \beta_1 x + \beta_2 x^2$

Linear Regression Format

- The general form of a linear model is:

$$\vec{Y} = W\vec{X}$$

- $\vec{X}$ is an N dimensional vector of features (indepedent variables; x_i, x_i^2, $x_i x_j$ etc.)
- $\vec{Y}$ is an M dimensional vector of responses (dependent variables)
- W is a design matrix which relates the regressors (indepdent variables) to the observables (dependent variables).

Weights and Biases

- The 0th component of the design matrix often incorporates the 0 reponse variable.
- This is the “intercept” of the model and when designed this way the feature vector is prepended with a 1 (a valid constant regressor).
- This can also be taken out and referred to as the biases:

  $$Y = WX + b$$

  .
- Weights and biases are a common way to refer to the parameters of the model.
- Biases indicates how biased each response variable is.

Linear Regression Error

- We need a procedure to perform the fit i.e. we need an error function.
- We choose the error between an observered regressor Y_j predicted regressor $\hat{Y}_j = WX_n + B$ as the least squared error:

$$e_j = \mathsf{sum}((\hat{Y}_j - Y_j).^2)$$

- The total error which we want to minimise is:

$$E = \sum_j e_j$$

$$E = \sum_j \mathsf{sum}(WX_j - Y_j).^2$$

Minimise the error.

- Consider the minimisation problem with just one data point.
- This has a quadratic function form $f(x, y) = (Wx - y)^2$.
- We can reliably get to the "bottom" of a quadratic using basic calculus setting the gradient to zero.
- The gradient here is easy to calculate: $2(Wx - y)(Wx - y)^T$

Total error

- The total error is linear in each of the data points: we can just sum up the minmima simultaneously.
- For a matrix of feature vectors $\mathbf{X}$ and responses $\mathbf{Y}$ we have:

$$E = (\mathbf{X}W - \mathbf{Y})^2$$

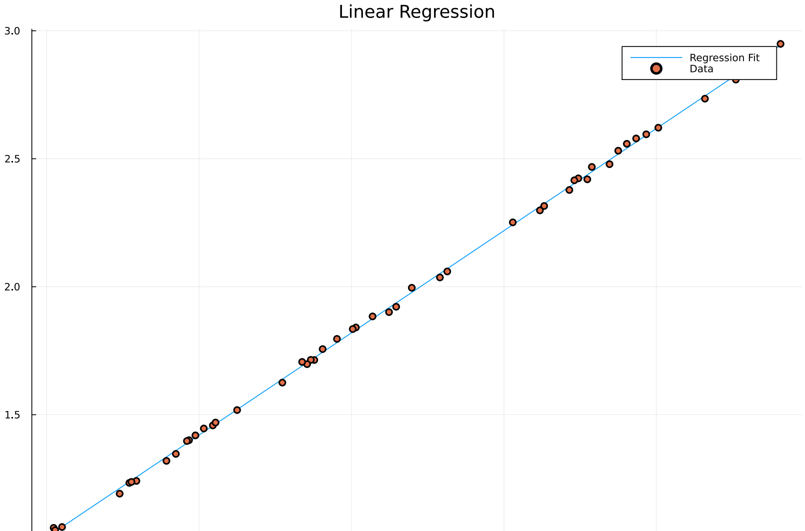
$$\frac{dE}{dw} = 2(\mathbf{X}w - \mathbf{W})(\mathbf{X}W - \mathbf{Y})^T = 0$$

$$W = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

- This is the optimal weights to minimise the mean squared error.

Example

```
x = hcat(ones(50), rand(50))';
y = [1 2] * x .+ 0.04 * rand(1, size(x)[2]);
W = (inv(x*x') * (x*y'))';
```



Nonlinear Regression

- Nonlinear regression allows the model to be non-linear in the parameters e.g.

$$f(x; \alpha, \beta) = \frac{x^n}{\alpha x^n + \beta^2}$$

.

- Nonlinear regression is generally more difficult to interpret: careful with goodness of fits.
- A common non-linear function we want to fit is the logistic function.

Logistic Function

- The logistic function is the solution to logistic equation often parameterised as:

$$p(x; \mu, \sigma) = \frac{1}{1 + \exp\left(\frac{\mu - x}{\sigma}\right)}$$

- It typically is interpreted to give a probability that is used to classify a variable.
- It comes up often in nature e.g. the firing response of a neuron.
- The logit is the inverse of the logistic function.

Logistic Regression

- Logistic regression attempts to estimate the parameters of the logistic function.
- Suppose $y \in \{0, 1\}$ and p(x) gives the probability of y. The cross entropy of the ith datapoint is defined as:

$$-y_i \ln(p(x_i)) - (1 - y_i) \ln(1 - p(x_i))$$

- It is zero if and only if all predictions are corrected. Otherwise it measures the entropy (think, disarray) between the two distributions.
- The parameters that minimise the cross-entropy are the best fit and are solved numerically through *gradient descent*; covered later.

Regression, Classification, and Generation

- Suppose that we have fitted our statistical model. The utility is in its *prediction*.
- The predictions are all technically regressions but are thought of in three categeries: regression, classification, and generation.
- Regression is the model output given some known choice of independent variable.
- Classification is the discrete model ouput (often the maximum probability) representing a category.
- Generation is the model output for some random (unseen) variable in the input space.

Data Preprocessing

- Often we find ourselves *exploring* data before we generate models on it.
- There are many useful techniques that can be applied to data.
- Noise reduction is an example of *data cleaning*.
- Clustering is a form of *unsupervised learning*.
- Principal Component Analysis is a form of *dimensionality reduction*.

Clustering

- Clustering involves partitioning a dataset into a series of different groups.
- This is generally done without labels and involves simple operations on the raw data.
- Clustering algorithms are themselves powerful statistical models but they wont be covered in this course.
- Common algorithims: `k-means` and `t-sne`.

Dimensionality Reduction

- Often data is very high-dimensional but these dimensions dont convey much information.
- Dimensionality reduction is a change of variables that captures most of the information with just a few variables.
- This can drastically simplify models.

Principal Component Analysis

- Principal Component Analysis is a very popular dimension reduction technique.
- It works by transforming the variables into a maximal variance encoding.
- This is achieved by eigenvalue decomposition of the covariance matrix.
- The principal eigenvector contains the most variance, and so on.

Learning

- Learning in the context of statistics refers to correctly parameterising a model.
- It is generally an iterative process where data is repeatedly presented to a learning algorithm.
- It is usually described as unsupervised when the data has not been “tagged” and supervised when it has.
- In supervised learning we know the “right” answer and can therefore construct a reasonable sense of error e.g. least squares.
- Clustering is unsupervised; linear regression is supervised.

Deep Learning?

- This has been a lot of general statistics. What does it have to do with Deep Learning?
- At the heart of all machine learning is distribution matching.
- Deep learning is just a powerful class of general non-linear statistical models.
- The model format takes inspiration from a physical structure (like most models) - the brain.
- We will go over this physical inspiration in the following lecture.

Summary

- Data is the entry point of all sciences.
- Models are precise and experimentally falsifiable relationships between data.
- Models are used for prediction and explanation.
- Deep Learning models are complex non-linear statistical models inspired by the brain.