machine learning basics

- ML can be applied to a wide variety of problems like classification, regression, transcription, machine translation, structured output (image segmentation for example), anomaly detection, synthesis and sampling from a probability distribution, imputation of missing values, denoising, density/pdf estimation, etc.
- Notice that some of these require the ML algorithm to capture the densities either explicitly or implicitly. This is a promising area to pursue and often falls under unsupervised learning.
- the ability of ML algorithms to perform well on previously unobserved inputs is called generalization.
- an estimator is said to be biased when a statistical estimation algorithm's expected estimate of a quantity is not equal to the true quantity. here expected means the formal meaning of the word "expectation".
- What separates machine learning from optimization is that we want the generalization error, also called the test error, to be low as well.
- How can we affect performance on the test set when we can observe only the training set? The field of statistical learning theory provides some answers.
- The factors determining how well a machine learning algorithm will perform are its ability to
  - 1. Make the training error small → underfitting problem
  - 2. Make the gap between training and test error small → overfitting problem
- Underfitting occurs when the model is not able to obtain a sufficiently low error value on the training set. Overfitting occurs when the gap between the training error and test error is too large.
- One way to control the capacity of a learning algorithm is by choosing its hypothesis space, the set of functions that the learning algorithm is allowed to select as being the solution.
- While coming up with a ML algorithm, you need:
  - right capacity for task at hand
  - enough, right data
  - a good optimizer: a good way to find the right function often when training is run
- Occam's razor says "among competing hypotheses, choose the simplest one".
- VC dimension is defined as being the largest possible value of m for which there exists a training set of m different x points that the classifier can label arbitrarily.
- the discrepancy between training error and generalization error is bounded from above by a quantity that grows as the model capacity grows but shrinks as the number of training examples increases.
- VC dimensions are rarely used in practice when working with deep learning algorithms.
- Parametric models learn a function described by a parameter vector whose size is finite and fixed before any data is observed. Nonparametric models have no such limitation.
- we can also design practical nonparametric models by making their complexity a function of the training set size. One example of such an algorithm is nearest neighbor regression.
- The ideal model is an oracle that simply knows the true probability distribution that generates the data. Even such a model will still incur some error on many problems, because there may still be some noise in the distribution. The error incurred by an oracle making predictions from the true distribution p(x, y) is called the Bayes error.
- it is possible for the model to have optimal capacity and yet still have a large gap between training and generalization errors. In this situation, we may be able to reduce this gap by gathering more training

examples.

- The no free lunch theorem for machine learning states that, averaged over all possible data-generating distributions, every classification algorithm has the same error rate when classifying previously unobserved points.
- Regularization: We can also give a learning algorithm a preference for one solution over another in its hypotheses space.
- Expressing preferences for one function over another is a more general way of controlling a model's capacity than including or excluding members from the hypothesis space.
- Regularization is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.
- you should have a separate set from the test set to choose good hyper parameters. this is called validation set.
- In practice, when the same test set has been used repeatedly to evaluate performance of different algorithms over many years, and especially if we consider all the attempts from the scientific community at beating the reported state-of-the-art performance on that test set, we end up having optimistic evaluations with the test set as well. Benchmarks can thus become stale and then do not reflect the true field performance of a trained system.
- it is still common practice to use them to declare that algorithm A is better than algorithm B only if the confidence interval of the error of algorithm A lies below and does not intersect the confidence interval of algorithm B.
- if you have small amount of data, use k-fold cross validation. On trial i, the i-th subset of the data is used as the test set, and the rest of the data is used as the training set. At the end, you could train on the whole data.
- A point estimator or statistic is any function of the data:
  $\hat{\theta} = g(x_1, ...x_m)$.
  The definition does not require that g return a value that is close to the true θ or even that the range of g be the same as the set of allowable values of θ.
- Bias of an estimator is $E(\hat{\theta}) - \theta$. Asymptotically unbiased means as you feed an infinite amount of data, bias approaches 0.
- The sample variance estimator for the gaussian distribution is biased. proof:

$$\begin{aligned}\mathbb{E}[\hat{\sigma}_m^2] =& \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m}\left(x^{(i)} - \hat{\mu}_m\right)^2\right] \\ =& \frac{m-1}{m}\sigma^2\end{aligned}$$

- While unbiased estimators are clearly desirable, they are not always the "best" estimators. As we will see we often use biased estimators that possess other important properties, like low variance in the estimate.
- Taking advantage of the central limit theorem, which tells us that the mean will be approximately distributed with a normal distribution, we can use the standard error to compute the probability that the true expectation falls in any chosen interval.
- The relationship between bias and variance is tightly linked to the machine learning concepts of capacity, underfitting and overfitting.

- Consistency ensures that the bias induced by the estimator diminishes as the number of data examples grows.
- One way to interpret maximum likelihood estimation is to view it as minimizing the KL divergence between the empirical distribution p_{data}, defined by the training set and the model distribution.
- Minimizing this KL divergence corresponds exactly to minimizing the cross- entropy between the distributions.
- Consistent estimators can differ in their statistical efficiency, meaning that one consistent estimator may obtain lower generalization error for a fixed number of samples m, or equivalently, may require fewer examples to obtain a fixed level of generalization error.
- When the number of examples is small enough to yield overfitting behavior, regularization strategies such as weight decay may be used to obtain a biased version of maximum likelihood that has less variance when training data is limited.
- the frequentist perspective is that the true parameter value θ is fixed but unknown, while the point estimate $\hat{\theta}$ is a random variable on account of it being a function of the dataset (which is seen as random).
- The Bayesian perspective on statistics is quite different. The Bayesian uses probability to reflect degrees of certainty in states of knowledge. The dataset is directly observed and so is not random. On the other hand, the true parameter θ is unknown or uncertain and thus is represented as a random variable.
- The prior has an influence by shifting probability mass density towards regions of the parameter space that are preferred a priori. In practice, the prior often expresses a preference for models that are simpler or more smooth.
- Bayesian methods typically generalize much better when limited training data is available but typically suffer from high computational cost when the number of training examples is large.
- The Bayesian estimate provides a covariance matrix, showing how likely all the different values of w are, rather than providing only the estimate μ.
- One common reason for desiring a point estimate is that most operations involving the Bayesian posterior for most interesting models are intractable, and a point estimate offers a tractable approximation. Rather than simply returning to the maximum likelihood estimate, we can still gain some of the benefit of the Bayesian approach by allowing the prior to influence the choice of the point estimate. One rational way to do this is to choose the maximum a posteriori (MAP) point estimate.

$$\boldsymbol{\theta}_{\text{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta} \mid \boldsymbol{x}) = \arg\max_{\boldsymbol{\theta}} \log p(\boldsymbol{x} \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}).$$

- As with full Bayesian inference, MAP Bayesian inference has the advantage of leveraging information that is brought by the prior and cannot be found in the training data.
- Many regularized estimation strategies, such as maximum likelihood learning regularized with weight decay, can be interpreted as making the MAP approximation to Bayesian inference. This view applies when the regularization consists of adding an extra term to the objective function that corresponds to log p(θ).
- This ability of PCA to transform data into a representation where the elements are mutually uncorrelated is a very important property of PCA. It is a simple example of a representation that attempts to disentangle the unknown factors of variation underlying the data. In the case of PCA, this

disentangling takes the form of finding a rotation of the input space (described by W) that aligns the principal axes of variance with the basis of the new representation space.

- The most common cost function is the negative log-likelihood, so that minimizing the cost function causes maximum likelihood estimation.
- The phenomenon is known as the curse of dimensionality: Of particular concern is that the number of possible distinct configurations of a set of variables increases exponentially as the number of variables increases.
- Among the most widely used of these implicit “priors” is the smoothness prior, or local constancy prior. This prior states that the function we learn should not change very much within a small region.
- While the k -nearest neighbors algorithm copies the output from nearby training examples, most kernel machines interpolate between training set outputs associated with nearby training examples. An important class of kernels is the family of local kernels, where k(u,v) is large when u = v and decreases as u and v grow further apart from each other. A local kernel can be thought of as a similarity function that performs template matching, by measuring how closely a test example x resembles each training example x_i . Much of the modern motivation for deep learning is derived from studying the limitations of local template matching and how deep models are able to succeed in cases where local template matching fails → presumably because weights of connections can break this property, which is a good thing?
- The key insight is that a very large number of regions, such as $O(2^k)$, can be defined with $O(k)$ examples, so long as we introduce some dependencies between the regions through additional assumptions about the underlying data-generating distribution. In this way, we can actually generalize nonlocally.
- Many different deep learning algorithms provide implicit or explicit assumptions that are reasonable for a broad range of AI tasks in order to capture these advantages.
- The core idea in deep learning is that we assume that the data was generated by the composition of factors, or features, potentially at multiple levels in a hierarchy.
- The exponential advantages conferred by the use of deep distributed representations counter the exponential challenges posed by the curse of dimensionality.
- Manifolds: in machine learning it tends to be used more loosely to designate a connected set of points that can be approximated well by considering only a small number of degrees of freedom, or dimensions, embedded in a higher-dimensional space. Each dimension corresponds to a local direction of variation.
- Manifold learning algorithms surmount this obstacle by assuming that most of R n consists of invalid inputs, and that interesting inputs occur only along a collection of manifolds containing a small subset of points, with interesting variations in the output of the learned function occurring only along directions that lie on the manifold, or with interesting variations happening only when we move from one manifold to another.

  Some intuition for interesting inputs occurring only along a collection of manifolds containing a small subset of points:

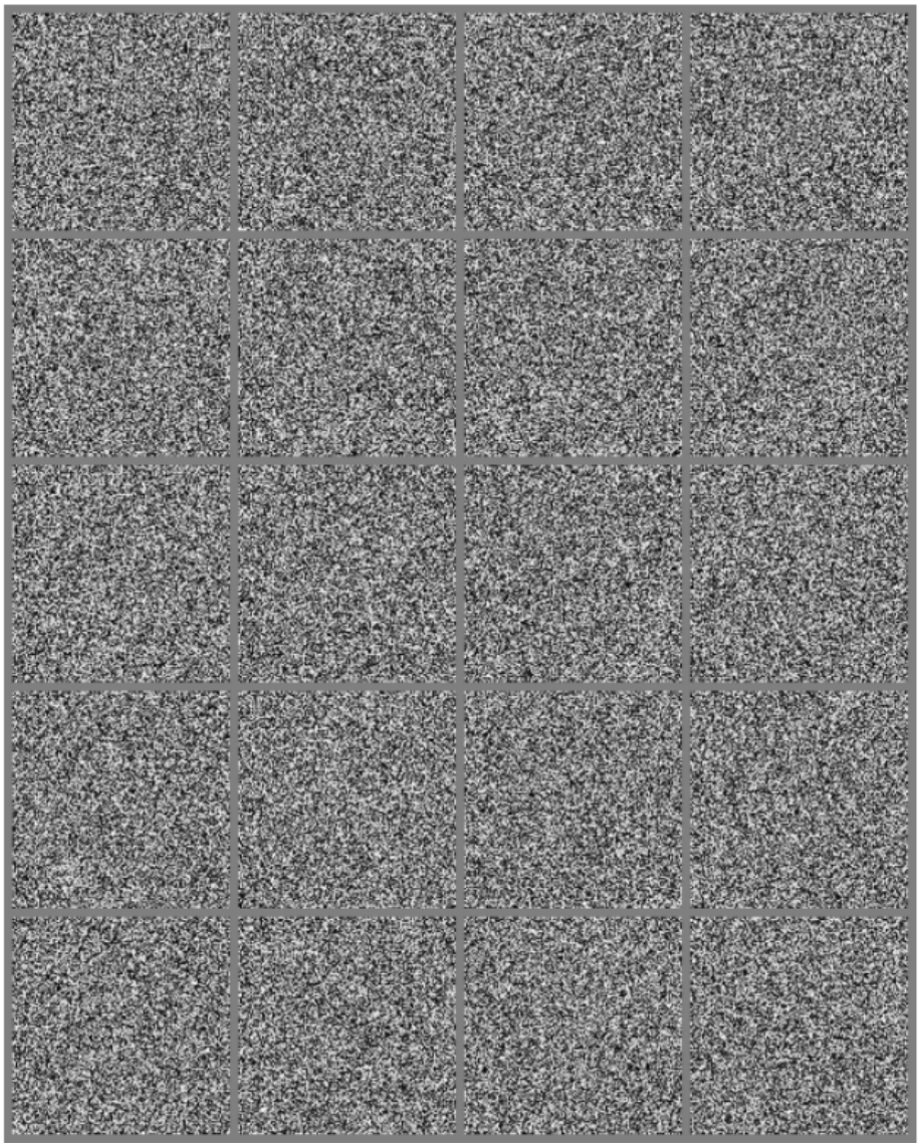

Figure 5.12: Sampling images uniformly at random (by randomly picking each pixel according to a uniform distribution) gives rise to noisy images. Although there is a nonzero probability of generating an image of a face or of any other object frequently encountered in AI applications, we never actually observe this happening in practice. This suggests that the images encountered in AI applications occupy a negligible proportion of the volume of image space.

-