Deep Learning

AlejandroMllo

This document serves as a very brief summary of the topics covered in each chapter of the book Deep Learning [1].

Disclaimer: This document is completely extracted from [1], the author does not attribute any ownership over the material.

5 Machine Learning Basics

- ML Algorithm: algorithm that is able to learn from data.
- Learning is the means of attaining the ability to perform a task.
- Some common ML tasks: classification, classification with missing inputs, regression, transcription, machine translation, structured output, anomaly detection, synthesis and sampling, imputation of missing values, denoising, density estimation or probability mass function estimation.
- Performance measure: quantitative measure (specific to the task) of the abilities of a ML algorithm.
- Unsupervised learning algorithms experience a dataset containing many features, then learn useful properties of the structure of this dataset.
- Supervised learning algorithms experience a dataset containing features, but each example is also associated with a label or target.
- Reinforcement learning algorithms interact with an environment, so there is a feedback loop between the learning system and its experiences.
- Design matrix: matrix containing a different example in each row; each column corresponds to a different feature.
  The vector $\boldsymbol{y}$, provides the label y_i to example i.
- Parameters (weights): values that control the behavior of the system.
- Linear Regression example:
  - Task: to predict y from $\boldsymbol{x}$ by outputting $\hat{y} = \boldsymbol{w}^\top \boldsymbol{x}$.
  - Performance measure: mean square error of the model on the test set.

    $$\mathtt{MSE}_{\mathtt{test}} = \frac{1}{m} \sum_i (\hat{\boldsymbol{y}}^{(\mathtt{test})} - \boldsymbol{y}^{(\mathtt{test})})_i^2$$

  - The objective is to design an algorithm that will improve the weights $\boldsymbol{w}$ in a way that reduces $\mathtt{MSE}_{\mathtt{test}}$ when the algorithm is allowed to gain experience by observing a training set $(\boldsymbol{X}^{(\mathtt{train})}, \boldsymbol{y}^{(\mathtt{train})})$. The way of doing this, it to minimize the MSE on the training set (solve for where its gradient is zero).
  - The idea is to minimize the training error, but measure the performance based on the test error.
  - The intercept term b of an affine function is often called the bias parameter.
- Generalization: the ability to perform well on previously unobserved inputs.
- Generalization/Test error: the expected value of the error on a new input.
- ML assumes that the datasets are independent and identically distributed.
- The factors determining how well a ML algorithm performs are its ability to make the training error small and make the gap between training and test error small.
- Underfitting: the model is not able to obtain a sufficiently low error value on the training set.
- Overfitting: the gap between the training error and test error is too large.
- Capacity: the model's ability to fit a wide variety of functions. One way to change it, is to change the number of input features it has and simultaneously add new parameters associated with those features.

- Hypothesis space: the set of functions that the learning algorithm is allowed to select as being the solution.
- Determine the capacity of a deep learning algorithm is difficult because the effective capacity is limited by the capabilities of the optimization algorithm.
- The no free lunch theorem for ML states that, averaged over all possible data-generating distributions, every classification algorithm has the same error rate when classifying previously unobserved points.
- The behavior of an algorithm is also affected by the specific identity of the functions in its hypothesis space.
- It is possible to regularize a model that learns a function $f(\boldsymbol{x}; \theta)$ by adding a penalty called a regularizer to the cost function. Regularization is any modification made to a learning algorithm that is intended to reduce its generalization error but not its training error.
- 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.
- Hyperparameters: settings used to control the algorithm's behavior.
- Point estimation: the attempt to provide the single 'best' prediction of some quantity of interest.
- Function estimation: approximatting f with a model or estimate $\hat{f}$.
- The bias of an estimator is defined as $\text{bias}(\hat{\boldsymbol{\theta}_m}) = \mathbb{E}(\hat{\boldsymbol{\theta}_m}) - \boldsymbol{\theta}$. It measures the expected deviation from the true value of the function or parameter.
- The variance of an estimator is simply the variance: $\text{Var}(\hat{\theta})$. It measures the deviation from the expected estimator value that any particular sampling of the data is likely to cause.
  The standard error, denoted $\text{SE}(\hat{\theta})$, is the square root of the variance.
- The generalization error is often estimated computing the sample mean of the error on the test set.
- Desirable estimators are those with small MSE and these are the estimators that manage to keep their bias and variance somewhat in check.
- Consistency: as the number of data points m in the dataset increases, the point estimates converge to the true value of the corresponding parameters. It ensures that the bias induced by the estimator diminishes as the number of data examples grows.
- Maximum Likelihood Estimation: minimizes the dissimilarity between the empirical distribution $\hat{p}_{data}$, defined by the training set and the model distribution, with the degree of dissimilarity between the two measured by the KL divergence. The KL divergence is given by

  $$D_{KL}(\hat{p}_{data}||p_{model}) = \mathbb{E}_{x\sim\hat{p}_{data}}[\log \hat{p}_{data}(\boldsymbol{x}) - \log p_{model}(\boldsymbol{x})]$$

  When training the model, it is only needed to minimize $-\mathbb{E}_{x\sim\hat{p}_{data}}[\log \hat{p}_{data}(\boldsymbol{x})]$.
- Bayesian statistics: considers all possible values of $\boldsymbol{\theta}$ when making a prediction. It uses probability to reflect degrees of certainty in states of knowledge.
  - The knowledge of $\boldsymbol{\theta}$ is represented using the prior probability distribution, $p(\boldsymbol{\theta})$.
  - To recover the effect of data on what is believed about $\boldsymbol{\theta}$:

    $$p(\boldsymbol{\theta}|x^{(1)}, \cdots, x^{(m)}) = \frac{p(x^{(1)}, \cdots, x^{(m)}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(x^{(1)}, \cdots, x^{(m)})}$$

- Support Vector Machine (SVM): supervised learning algorithm that outputs a class identity.
- k-nearest neighbors: family of supervised learning techniques used for classification or regression.
- Principal Component Analysis learns a representation that has lower dimensionality than the original input. It also learns a representation whose elements have no linear correlation with each other. This is a first step toward the criterion of learning representations whose elements are statistically independent. To achieve full independence, a representation learning algorithm must also remove the nonlinear relationships between variables.
- Stochastic Gradient Descent (SGD): nearly all of deep learning is powered by this algorithm.
  - The insight of SGD is that the gradient is an expectation. It can be approximately estimated using a small set of samples.
  - Each step of the algorithm samples a minibatch of examples $\mathbb{B} = \{\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(m')}\}$ drawn uniformly from the training set.
  - The estimate of the gradient is formed as:

    $$\boldsymbol{g} = \frac{1}{m'}\nabla_{\boldsymbol{\theta}}\sum_{i=1}^{m'} L(\boldsymbol{x}^{(i)}, y^{(i)}, \boldsymbol{\theta})$$

    using examples from the minibatch $\mathbb{B}$.

- The algorithm then follows the estimated gradient downhill:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \epsilon \boldsymbol{g}$$

where ϵ is the learning rate.

- The recipe, of most, deep learning algorithm can be described as: combine a specification of a dataset, a cost function, an optimization procedure and a model.
- The Curse of Dimensionality: many ML algorithms become exceedingly difficult when the number of dimensions in the data is high. The number of possible distinct configuration of a set of variables increases exponentially as the number of variables increases.
- Smoothness prior or local constancy prior: this prior states that the learned function should not change very much within a small region.
- The core idea in deep learning is that is assumes that the data was generated by the composition of factors, or features, potentially at multiple levels in a hierarchy.
- Manifold: connected region. Mathematically, it is a set of points associated with a neighborhood around each point. From any given point, it locally appears to be a Euclidean space. - In ML it tends to be used 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.
- Manifold learning algorithms assume that most of $\mathbb{R}^n$ 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 moving from one manifold to another.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.