2. Linear Classifier and Regressor

2.1 Linear Function

In calculus and related areas, a linear function is a function whose graph is a straight line, that is, **a polynomial function of degree zero or one**. For distinguishing such a linear function from the other concept, the term affine function is often used.

In mathematics, a linear function (linear map) is a mapping $V \to W$ between two vector spaces that preserves the operations of vector addition and scalar multiplication.

If a linear map is a bijection then it is called a linear isomorphism. In the case where $V = W$, a linear map is called a (linear) endomorphism. Sometimes the term linear operator refers to this case, but the term "linear operator" can have different meanings for different conventions: for example, it can be used to emphasise that V and W are real vector spaces (not necessarily with $V = W$), or it can be used to emphasise that V is a function space, which is a common convention in functional analysis. Sometimes the term linear function has the same meaning as linear map, while in analysis it does not.

A linear map from V to W always maps the origin of V to the origin of W. Moreover, it maps linear subspaces in V onto linear subspaces in W (possibly of a lower dimension); for example, it maps a plane through the origin in V to either a plane through the origin in W, a line through the origin in W, or just the origin in W. Linear maps can often be represented as matrices, and simple examples include rotation and reflection linear transformations.

阅读后结合上课内容需对线性函数/线性映射有充分的理解，能够从输入输出的角度判断线性关系，深刻理解线性函数对点乘和加法的贯通性质。

2.2 Linear Classifier

The goal of statistical classification is to use an object's characteristics to identify which class (or group) it belongs to. A linear classifier achieves this by making a classification decision based on the value of a linear combination of the characteristics (features). An object's characteristics are also known as feature values and are typically presented to the machine in a vector called a feature vector ($[x_1, x_2, \ldots, x_m]$). Such classifiers work well for practical problems such as document classification, and more generally for problems with many variables (features), reaching accuracy levels comparable to non-linear classifiers (like human decision, out of the scope of this course) while taking less time to train and use.

If the input feature vector to the classifier is a real vector $\vec{x}$, then the output score is

$$y = f(\vec{w} \cdot \vec{x}) = f\left(\sum_j w_j x_j\right),$$

, where $\vec{w} = [w_1, w_2, \ldots, w_m]$ is a real vector of weights and f is a function that converts the dot product of the two vectors into the desired output. (In other words, $\vec{w}$ is a one-form or linear functional mapping $\vec{x}$ onto $\mathbb{R}$.) The weight vector $\vec{w}$ is learned from a set of labelled training samples. Often f is a threshold function, which maps all values of $\vec{w} \cdot \vec{x}$ above a certain threshold to the first class and all other values to the second class; *e.g.*,

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w}^T \cdot \mathbf{x} > \theta, \\ 0 & \text{otherwise} \end{cases}$$

The superscript T indicates the transpose and θ is a scalar threshold. A more complex f might give the probability that an item belongs to a certain class.

For a two-class classification problem, one can visualise the operation of a linear classifier as splitting a high-dimensional input space with a hyperplane: all points on one side of the hyperplane are classified as "yes", while the others are classified as "no".

A linear classifier is often used in situations where the speed of classification is an issue, since it is often the simplest classifier, especially when $\vec{x}$ is sparse. Also, linear classifiers often work very well when the number of dimensions in $\vec{x}$ is large, as in document classification, where each element in $\vec{x}$ is typically the number of occurrences of a word in a document (see document-term matrix). In such cases, the classifier should be well-regularised.

阅读后结合上课内容需对线性分类器的定义有直观且深入的理解，明白其为最简单分类模型的原因，从几何的角度展开想象。

2.3 Loss Function

In mathematical optimisation and decision theory, a loss function or cost function (sometimes also called an error function) is a function that maps an event or values of one or more variables onto a real number intuitively representing some "cost" associated with the event. An optimisation problem seeks to minimise a loss function. An objective function is either a loss function or its opposite, in which case it is to be maximised.

In statistics, typically a loss function is used for parameter estimation, and the event in question is some function of the difference between estimated and true values for an instance of data. In classification, it is the penalty for an incorrect classification of an example. In optimal control, the loss is the penalty for failing to achieve a desired value. In financial risk management, the function is mapped to a monetary loss.

This course covers the following typical loss functions, *i.e.*, **0-1 loss function, Hinge loss, Square loss, Exponential loss**. Please refer to Section 2.4 to Section 2.7 for more details.

阅读后结合上课内容需对损失函数（目标函数）的定义和直观解释有充分的了解和掌握。在章节 2.4到章节 2.7的阅读学习中反复观察与分析它们在图 2.1中体现出的特性。

2.4 0-1 Loss

In statistics and decision theory, a frequently used loss function is the 0-1 loss function

$$\mathscr{L}(\hat{y}, y) = \mathbb{I}(\hat{y} \neq y),$$

where $\mathbb{I}$ is the indicator function.

In mathematics, an indicator function or a characteristic function of a subset of a set is a function that maps elements of the subset to one, and all other elements of the set to zero. The indicator function of a subset A of a set X maps X to the two-element set $\{0, 1\}$; $\mathbb{I}_A(x) = 1$ if an element x in X belongs to A, and $\mathbb{I}_A(x) = 0$ if x does not belong to A.

阅读后结合上课内容进一步理解和掌握 0-1 损失的天然性意义。

2.5 Hinge Loss

$$\mathscr{L}(f(\vec{x}), y) = \max(0, 1 - yf(\vec{x})) = [1 - yf(\vec{x})]_+.$$

The hinge loss provides a relatively tight, convex upper bound on the 0–1 indicator function. Specifically, the hinge loss equals the 0–1 indicator function when $\operatorname{sgn}(f(\vec{x})) = y$ and $|yf(\vec{x})| \geq 1$. While the hinge loss function is both convex and continuous, it is not smooth (is not differentiable) at $yf(\vec{x}) = 1$. Consequently, the hinge loss function cannot be used with gradient descent methods or

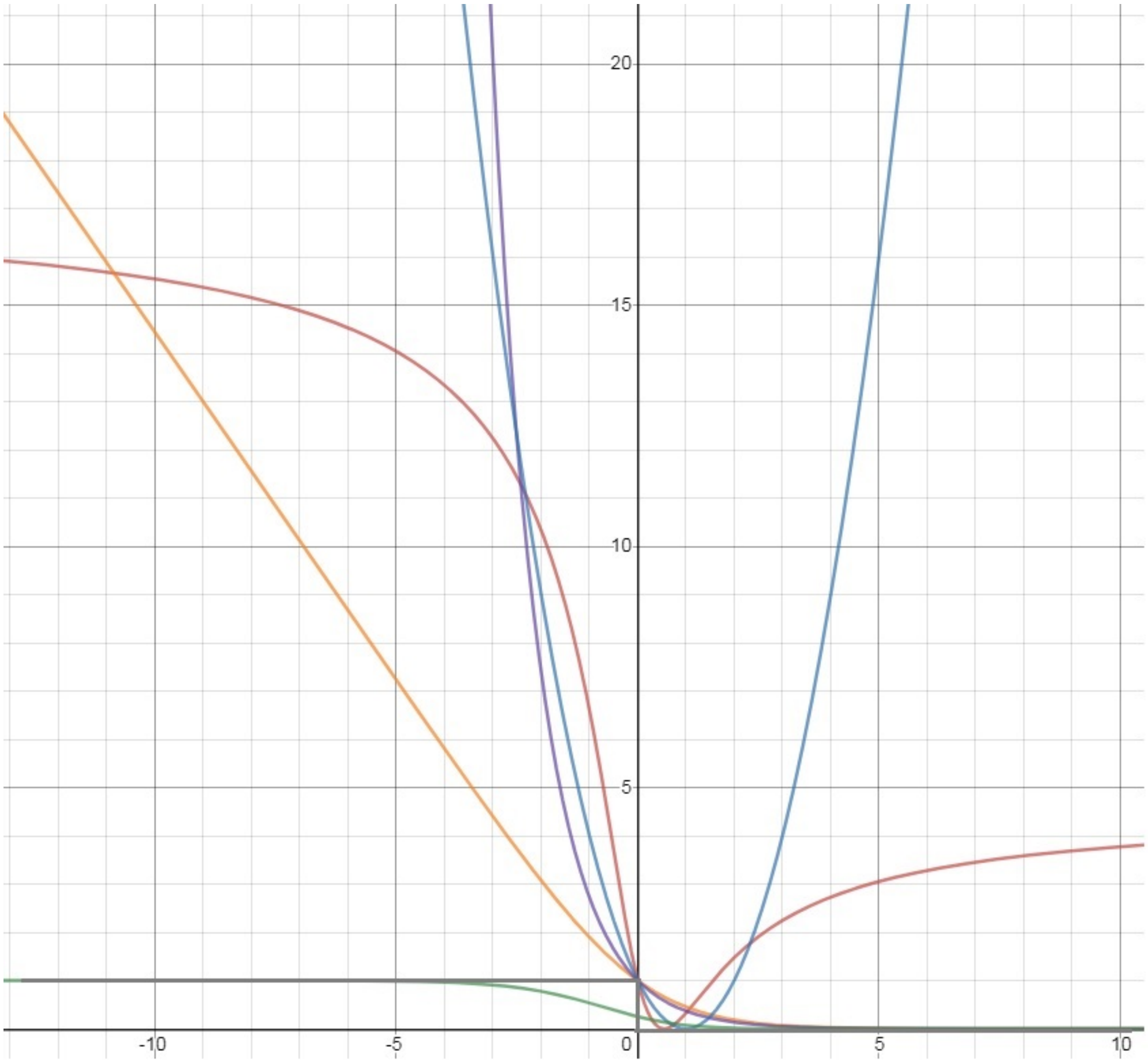

Figure 2.1: Illustration of typical losses: Zero-one loss (gray), Savage loss (green), Logistic loss (orange), Exponential loss (purple), Tangent loss (brown), Square loss (blue).

stochastic gradient descent methods which rely on differentiability over the entire domain. However, the hinge loss does have a subgradient at $yf(\vec{x}) = 1$, which allows for the utilisation of subgradient descent methods.

阅读后结合上课内容掌握 Hinge 损失，和它与 0-1 损失的关系。

2.6 Square Loss

$$\mathscr{L}(f(\vec{x}), y) = (f(\vec{x}) - y)^2$$

The square loss function is both convex and smooth. However, the square loss function tends to penalise outliers excessively, leading to slower convergence rates (with regards to sample complexity) than for the hinge loss functions. In addition, functions which yield high values of $f(\vec{x})$ for some $x \in X$ will perform poorly with the square loss function, since high values of $yf(\vec{x})$ will be penalised severely, regardless of whether the signs of y and $f(\vec{x})$ match.

A benefit of the square loss function is that its structure lends itself to easy cross validation of regularisation parameters. Specifically for Tikhonov regularisation, one can solve for the regularisation parameter using leave-one-out cross-validation in the same time as it would take to solve a single problem.

阅读后结合上课内容掌握 square 损失，它和 Hinge 损失的关系，以及它的优势和缺点。

2.7 Exponential Loss

$$\mathscr{L}(f(\vec{x}), y) = \exp -yf(\vec{x})$$

阅读后结合上课内容掌握 exponential 损失，通过分析其与 Hinge 损失、Square 损失在曲线上的差异，判断出其存在的优势和缺点。

2.8 Gradient Descent

In mathematics gradient descent (also often called steepest descent) is a first-order iterative optimisation algorithm for finding a local minimum of a differentiable function. The idea is to take repeated steps in the opposite direction of the gradient (or approximate gradient) of the function at the current point, because this is the direction of steepest descent. Conversely, stepping in the direction of the gradient will lead to a local maximum of that function; the procedure is then known as gradient ascent.

Gradient descent is generally attributed to Cauchy, who first suggested it in 1847. Hadamard independently proposed a similar method in 1907. Its convergence properties for non-linear optimisation problems were first studied by Haskell Curry in 1944, with the method becoming increasingly well-studied and used in the following decades.

Gradient descent is based on the observation that if the multi-variable (input is a vector) function $F(\mathbf{x})$ is defined and differentiable in a neighborhood of a point $\mathbf{a}$, then $F(\mathbf{x})$ decreases fastest if one goes from $\mathbf{a}$ in the direction of the negative gradient of F at $\mathbf{a}, -\nabla F(\mathbf{a})$. It follows that, if $\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$ for a small enough step size or learning rate $\gamma \in \mathbb{R}_+$, then $F(\mathbf{a_n}) \geq F(\mathbf{a_{n+1}})$. In other words, the term $\gamma \nabla F(\mathbf{a})$ is subtracted from $\mathbf{a}$ because we want to move against the gradient,

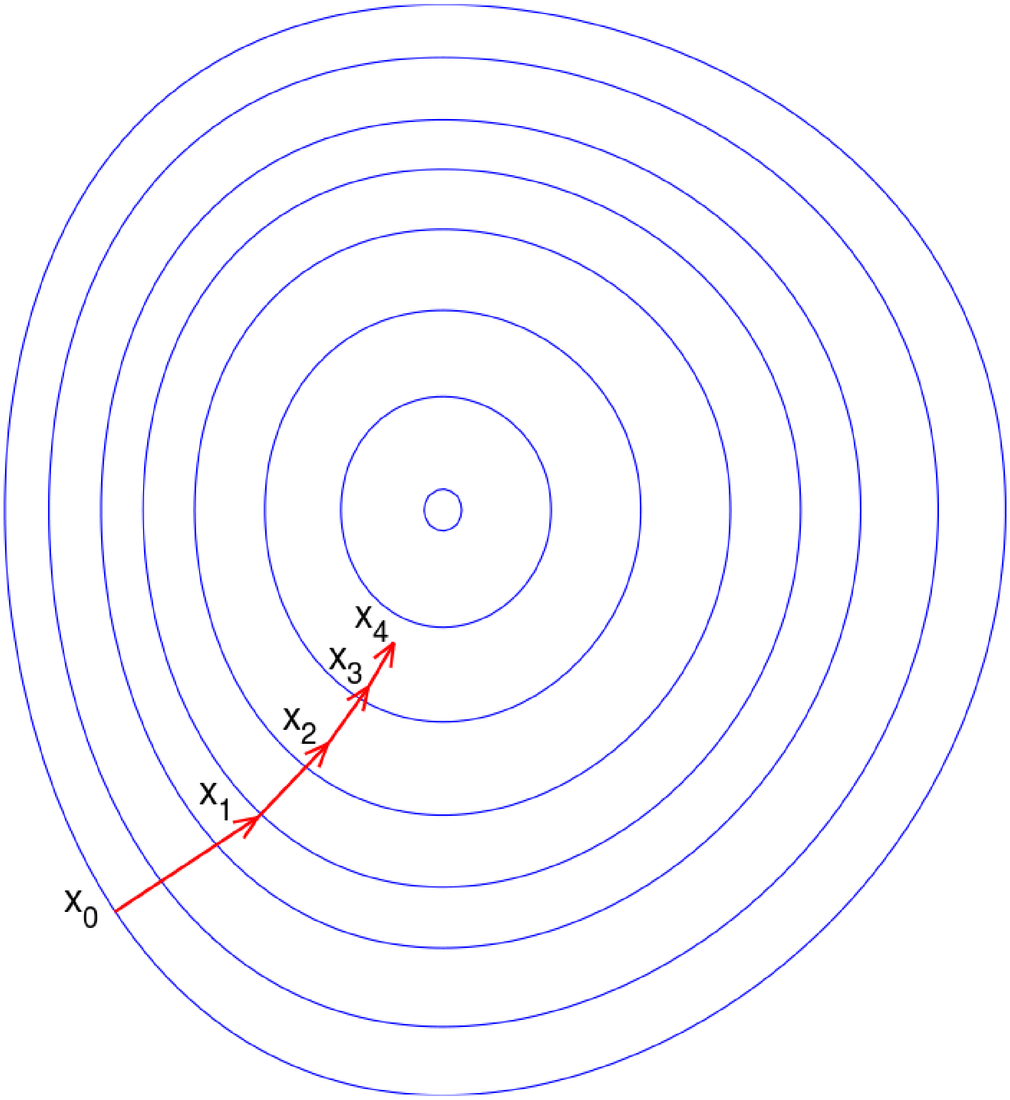

Figure 2.2: Illustration of gradient descent on a series of level sets

toward the local minimum. With this observation in mind, one starts with a guess $\mathbf{x}_0$ for a local minimum of F, and considers the sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$ such that $\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma_n \nabla F(\mathbf{x}_n),\ n \geq 0$. We have a monotonic sequence $F(\mathbf{x}_0) \geq F(\mathbf{x}_1) \geq F(\mathbf{x}_2) \geq \cdots$, so, hopefully, the sequence $(\mathbf{x}_n)$ converges to the desired local minimum. Note that the value of the step size γ is allowed to change at every iteration. With certain assumptions on the function F (for example, F convex and ∇F

Lipschitz) and particular choices of γ (e.g., chosen either via a line search that satisfies the Wolfe conditions, or the Barzilai–Borwein method shown as following),

$$\gamma_n = \frac{\left|\left(\mathbf{x}_n - \mathbf{x}_{n-1}\right)^T \left[\nabla F\left(\mathbf{x}_n\right) - \nabla F\left(\mathbf{x}_{n-1}\right)\right]\right|}{\left\|\nabla F\left(\mathbf{x}_n\right) - \nabla F\left(\mathbf{x}_{n-1}\right)\right\|^2}$$

convergence to a local minimum can be guaranteed. **When the function F is convex, all local minima are also global minima, so in this case gradient descent can converge to the global solution.**

This process is illustrated in Figure 2.2. Here, F is assumed to be defined on the plane (2D), and that its graph has a bowl shape. The blue curves are the contour lines, that is, the regions on which the value of F is constant. A red arrow originating at a point shows the direction of the negative gradient at that point. Note that the (negative) gradient at a point is orthogonal to the contour line going through that point. We see that gradient descent leads us to the bottom of the bowl, that is, to the point where the value of the function F is minimal.

阅读后结合上课内容掌握整个梯度下降法的前提、过程、结论。对计算过程有必要动手推导一遍，结合 Square 损失和线性分类器模型构建出完整的“模型-损失-优化”链条蓝色部分为拓展知识，解释了动态调整学习率的一种方式。 推荐此部分的外部参考资源于 B 站 https://www.bilibili.com/read/cv6595551

2.9 Inverse Problem

An inverse problem in science is the process of calculating from a set of observations the causal factors that produced them: for example, calculating an image in X-ray computed tomography, source reconstruction in acoustics, or calculating the density of the Earth from measurements of its gravity field. **It is called an inverse problem because it starts with the effects and then calculates the causes. It is the inverse of a forward problem, which starts with the causes and then calculates the effects.**

Inverse problems are some of the most important mathematical problems in science and mathematics because they tell us about parameters that we cannot directly observe. They have wide application in system identification, optics, radar, acoustics, communication theory, signal processing, medical imaging, computer vision, geophysics, oceanography, astronomy, remote sensing, natural language processing, machine learning, nondestructive testing, slope stability analysis and many other fields.

阅读后结合上课内容了解逆（inverse）问题和前向（forward）问题的概念。

2.10 Condition Number

In numerical analysis, the condition number of a function measures how much the output value of the function can change for a small change in the input argument. This is used to measure how sensitive a function is to changes or errors in the input, and how much error in the output results from an error in the input. Very frequently, one is solving the inverse problem: given $f(x) = y$, one is solving for x, and thus the condition number of the (local) inverse must be used. In linear regression the condition number of the moment matrix can be used as a diagnostic for multicollinearity.

The condition number is an application of the derivative, and is formally defined as the value of the asymptotic worst-case relative change in output for a relative change in input. The "function" is the solution of a problem and the "arguments" are the data in the problem. The condition number is frequently applied to questions in linear algebra, in which case the derivative is straightforward but the error could be in many different directions, and is thus computed from the geometry of the matrix. More generally, condition numbers can be defined for non-linear functions in several variables.

A problem with a low condition number is said to be well-conditioned, while a problem with a high condition number is said to be ill-conditioned. In non-mathematical terms, an ill-conditioned problem is one where, for a small change in the inputs (the independent variables) there is a large change in the answer or dependent variable. This means that the correct solution/answer to the equation becomes hard to find. The condition number is a property of the problem. Paired with the problem are any number of algorithms that can be used to solve the problem, that is, to calculate the solution. Some algorithms have a property called backward stability. In general, a backward stable algorithm can be expected to accurately solve well-conditioned problems. Numerical analysis textbooks give formulas for the condition numbers of problems and identify known backward stable algorithms.

As a rule of thumb, if the condition number $\kappa(A) = 10^k$, then you may lose up to k digits of accuracy on top of what would be lost to the numerical method due to loss of precision from arithmetic methods. However, the condition number does not give the exact value of the maximum inaccuracy that may occur in the algorithm. It generally just bounds it with an estimate (whose computed value depends on the choice of the norm to measure the inaccuracy).

阅读后结合上课内容理解条件数和病态问题的关系。蓝色部分为拓展知识。理解相关概念后可以加深对矩阵这一数学表征的进一步感悟。 推荐此部分的外部参考资源于 B 站 https://www.bilibili.com/video/BV1xk4y1C7dc?share_source=copy_web

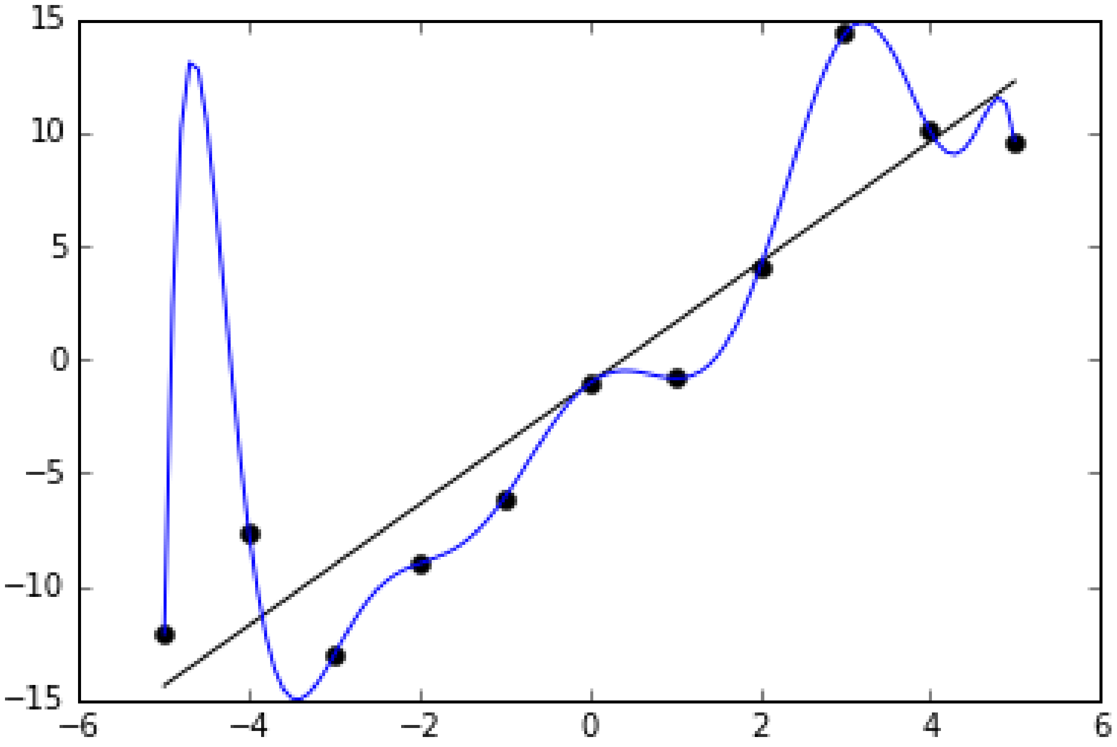

Figure 2.3: Noisy (roughly linear) data is fitted to a linear function and a polynomial function. Although the polynomial function is a perfect fit, the linear function can be expected to generalize better: if the two functions were used to extrapolate beyond the fitted data, the linear function should make better predictions.

2.11 Overfitting

In statistics, overfitting is "the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably". An overfitted model is a statistical model that contains more parameters than can be justified by the data. The essence of overfitting is to have unknowingly extracted some of the residual variation (i.e., the noise) as if that variation represented underlying model structure.

Underfitting occurs when a statistical model cannot adequately capture the underlying structure of the data. An under-fitted model is a model where some parameters or terms that would appear in a correctly specified model are missing. Under-fitting would occur, for example, when fitting a linear model to non-linear data. Such a model will tend to have poor predictive performance.

The possibility of over-fitting exists because the criterion used for selecting the model is not the same as the criterion used to judge the suitability of a model. For example, a model might be selected by maximizing its performance on some set of training data, and yet its suitability might be determined by its ability to perform well on unseen data; then over-fitting occurs when a model begins to "memorise" training data rather than "learning" to generalise from a trend.

As an extreme example, if the number of parameters is the same as or greater than the number of observations, then a model can perfectly predict the training data simply by memorising the data in its entirety. (For an illustration, see Figure 5.1.) Such a model, though, will typically fail severely when making predictions.

The potential for overfitting depends not only on the number of parameters and data but also the conformability of the model structure with the data shape, and the magnitude of model error compared to the expected level of noise or error in the data. Even when the fitted model does not have an excessive number of parameters, it is to be expected that the fitted relationship will appear to perform less well on a new data set than on the data set used for fitting (a phenomenon sometimes known as shrinkage). In particular, the value of the coefficient of determination will shrink relative to the original data.

To lessen the chance or amount of overfitting, several techniques are available (e.g., model comparison, cross-validation, regularisation, early stopping, pruning, Bayesian priors, or dropout). The basis of some techniques is either (1) to explicitly penalise overly complex models or (2) to test the model's ability to generalise by evaluating its performance on a set of data not used for training, which is assumed to approximate the typical unseen data that a model will encounter.

阅读后结合上课内容掌握过拟合和欠拟合的概念、原因、及解决办法。

2.12 Regularisation

In mathematics, statistics, finance, computer science, particularly in machine learning and inverse problems, regularisation is the process of adding information in order to solve an ill-posed problem or to prevent overfitting.

Regularisation can be applied to objective functions in ill-posed optimisation problems. The regularisation term, or penalty, imposes a cost on the optimisation function to make the optimal solution unique.

Independent of the problem or model, there is always a data term, that corresponds to a likelihood of the measurement and a regularisation term that corresponds to a prior. By combining both using Bayesian statistics, one can compute a posterior, that includes both information sources and therefore stabilises the estimation process. By trading off both objectives, one chooses to be more addictive to the data or to enforce generalisation (to prevent overfitting). There is a whole research branch dealing with all possible regularisation's. The work flow usually is, that one tries a specific regularisation and then figures out the probability density that corresponds to that regularisation to

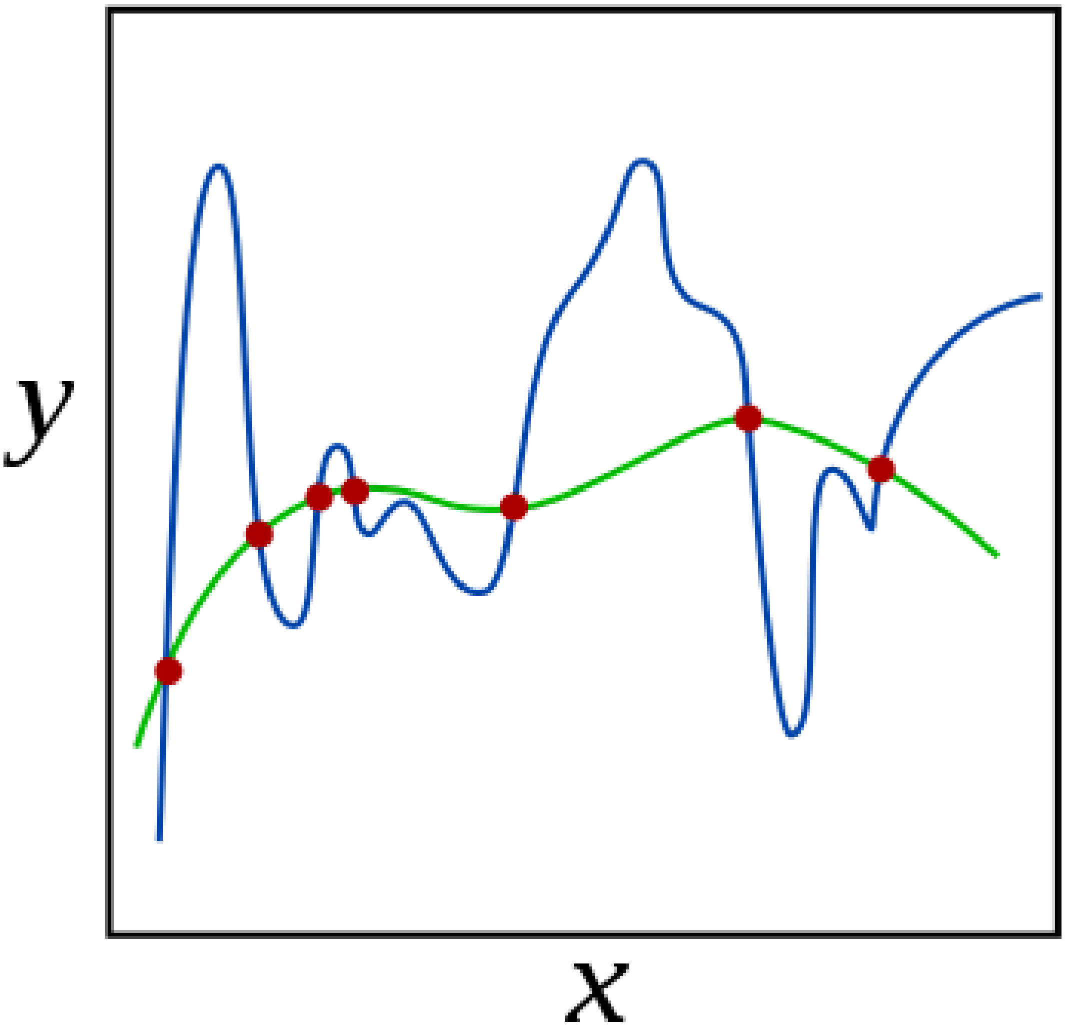

Figure 2.4: The green and blue functions both incur zero loss on the given data points. A learned model can be induced to prefer the green function, which may generalise better to more points drawn from the underlying unknown distribution, by adjusting λ, the weight of the regularisation term.

justify the choice. It can also be physically motivated by common sense or intuition.

In machine learning, the data term corresponds to the training data and the regularisation is either the choice of the model or modifications to the algorithm. It is always intended to reduce the generalisation error, i.e. the error score with the trained model on the evaluation set and not the training data.

Empirical learning of classifiers (from a finite data set) is always an underdetermined problem, because it attempts to infer a function of any x given only examples $x_1, x_2, \ldots x_n$.

A regularisation term (or regulariser) $R(f)$ is added to a loss function:

$$\min_f \sum_{i=1}^{n} V(f(x_i), y_i) + \lambda R(f)$$

where V is an underlying loss function that describes the cost of predicting $f(x)$ when the label is y, such as the square loss or hinge loss; and λ is a parameter which controls the importance of the regularisation term. $R(f)$ is typically chosen to impose a penalty on the complexity of f. Concrete notions of complexity used include restrictions for smoothness and bounds on the vector space norm.

A theoretical justification for regularisation is that it attempts to impose Occam's razor on the solution (as depicted in the figure above, where the green function, the simpler one, may be preferred). From a Bayesian point of view, many regularisation techniques correspond to imposing certain prior distributions on model parameters.

Regularisation can serve multiple purposes, including learning simpler models, inducing models to be sparse and introducing group structure into the learning problem.

The same idea arose in many fields of science. A simple form of regularisation applied to integral equations (Tikhonov regularisation) is essentially a trade-off between fitting the data and reducing a norm of the solution. More recently, non-linear regularisation methods, including total variation regularisation, have become popular.

阅读后结合上课内容掌握正则的概念和意义，理解奥卡姆剃刀原理。

2.13 Norm

In mathematics, a norm is a function from a real or complex vector space to the non-negative real numbers that behaves in certain ways like the distance from the origin: it commutes with scaling, obeys a form of the triangle inequality, and is zero only at the origin. In particular, the Euclidean distance of a vector from the origin is a norm, called the Euclidean norm, or 2-norm, which may also be defined as the square root of the inner product of a vector with itself.

A pseudonorm or seminorm satisfies the first two properties of a norm, but may be zero for vectors other than the origin. A vector space with a specified norm is called a normed vector space. In a similar manner, a vector space with a seminorm is called a seminormed vector space.

Given a vector space X over a subfield F of the complex numbers $\mathbb{C}$, a norm on X is a real-valued function $p : X \rightarrow \mathbb{R}$ with the following properties, where $|s|$ denotes the usual absolute value of a scalar s:

(1) Subadditivity/Triangle inequality: $p(x+y) \leq p(x) + p(y)$ for all $x, y \in X$.

(2) Absolute homogeneity: $p(sx) = |s|\, p(x)$ for all $x \in X$ and all scalars s.

(3) Positive definiteness/Point-separating: for all $x \in X$, if $p(x) = 0$ then $x = 0$.

Because property (2) implies $p(0) = 0$, some authors replace property (3) with the equivalent condition: for every $x \in X$, $p(x) = 0$ if and only if $x = 0$. A seminorm on X is a function $p : X \to \mathbb{R}$ that has properties (1) and (2) so that in particular, every norm is also a seminorm (and thus also a sublinear functional). However, there exist seminorms that are not norms. Properties (1) and (2) imply that if p is a norm (or more generally, a seminorm) then $p(0) = 0$ and that p also has the following property:

Non-negativity: $p(x) \geq 0$ for all $x \in X$. Some authors include non-negativity as part of the definition of "norm", although this is not necessary.

阅读后结合上课内容了解什么是范数。

2.14 p-norm

L^p space Let $p \geq 1$ be a real number. The p-norm (also called ℓ_p-norm) of vector $\mathbf{x} = (x_1, \ldots, x_n)$ is

$$\|\mathbf{x}\|_p := \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}.$$

For $p = 1$, we get the taxicab norm, for $p = 2$ we get the Euclidean norm, and as p approaches ∞ the p-norm approaches the infinity norm or maximum norm: $\|\mathbf{x}\|_\infty := \max_i |x_i|$. The p-norm is related to the generalised mean or power mean. This definition is still of some interest for $0 \leq p \leq 1$, but the resulting function does not define a norm, because it violates the triangle inequality. What is true for this case of $0 \leq p \leq 1$, even in the measurable analog, is that the corresponding L^p class is a vector space, and it is also true that the function $\int_X |f(x) - g(x)|^p \, \mathrm{d}\mu$ (without pth root) defines a distance that makes $L^p(X)$ into a complete metric topological vector space. These spaces are of great interest in functional analysis, probability theory and harmonic analysis. However, aside from trivial cases, this topological vector space is not locally convex, and has no continuous non-zero linear forms. Thus the topological dual space contains only the zero functional. The partial derivative of the p-norm is given by

$$\frac{\partial}{\partial x_k} \|\mathbf{x}\|_p = \frac{x_k |x_k|^{p-2}}{\|\mathbf{x}\|_p^{p-1}}.$$

The derivative with respect to x, therefore, is

$$\frac{\partial \|\mathbf{x}\|_p}{\partial \mathbf{x}} = \frac{\mathbf{x} \circ |\mathbf{x}|^{p-2}}{\|\mathbf{x}\|_p^{p-1}}.$$

where denotes Hadamard product and $|\cdot|$ is used for absolute value of each component of the vector. For the special case of $p = 2$, this becomes

$$\frac{\partial}{\partial x_k}\|\mathbf{x}\|_2 = \frac{x_k}{\|\mathbf{x}\|_2},$$

or

$$\frac{\partial}{\partial \mathbf{x}}\|\mathbf{x}\|_2 = \frac{\mathbf{x}}{\|\mathbf{x}\|_2}.$$

阅读后结合上课内容掌握 p 范数的数学定义，深刻理解 1 范数和 2 范数。