Machine Learning Explained

Lecture Notes of CS385, SJTU, taught by Prof. Quanshi Zhang

Summarized and Written by Tony Fang (galaxies@sjtu.edu.cn, tony.fang.galaxies@gmail.com)

Chapter 3. Regularized Learning

3.1 Overfitting and Underfitting

Overfitting means "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 the variation represented underlying model structure. The main properties of overfitting can be concluded as:

- Well fit to training samples, but badly fit to testing samples;
- The model usually has much more parameters than necessary;
- The model considers feature noises as meaningful information.

Underfitting occurs when a statistical model cannot adequately capture the underlying structure of the data. An underfitted model is a model where some parameters of terms would appear in a correctly specified model are missing. Underfitting would occur, for example, when fitting a linear model to non-linear data. Such a model will tend to have poor predictive performance. The main properties of underfitting can be concluded as:

- Badly fit to both training samples and testing samples;
- The model is not flexible enough, *e.g.*, the parameters in the model are not enough; or the optimization method is not powerful enough;
- The model misses meaningful information.

The ideal model should fit well both to training samples and testing samples. In conclusion, the relationship among performance, training set and testing set performances is as follows.

Table 1. Overfitting, Underfitting and Ideal Model[1]

Performance	Training Set	Validation Set
Underfitting	Large Error	Irrelevant
Overfitting	Small Error	Large Error
Ideal (Good Generalization)	Small Error	Small Error

3.2 Ridge Regression

Given training samples $\{(X_i, y_i)\}_{i=1,2,\cdots,n}$, where X_i is a p-dimensional vector and y_i is the ground-truth. The ridge regression is based on a linear regression model $y = X\beta + \varepsilon$, where X is an $n \times p$ matrix consisting of X_i, Y is an $n \times 1$ vector of ground truth, and $\varepsilon \sim N(0, \sigma^2 I_n)$.

Instead of only using the square loss, the ridge regression defines the loss function as follows.

$$\mathcal{L}(\beta) = \|Y - X\beta\|^2 + \lambda\|\beta\|^2$$

where $\|Y - X\beta\|^2$ is the conventional square loss, $\lambda\|\beta\|^2$ is the regularization term penalizing the weight dimensions with large absolute values, which prevent the regression from being conducted based on a few feature dimensions and boosts the robustness of the model, and λ is a hyper-parameter of the model.

Thus, we want to find $\hat{\beta}_\lambda$ with minimal loss.

$$\hat{\beta}_\lambda = \arg\min_\beta \mathcal{L}(\beta) = \arg\min_\beta \|Y - X\beta\|^2 + \lambda\|\beta\|^2$$

The first order condition of the optimization problem is

$$\frac{\partial \mathcal{L}(\beta)}{\partial \beta} = 2X^T X\beta - 2X^T Y + 2\lambda\beta = 0$$

Hence, the estimate of β is

$$\hat{\beta}_\lambda = \left(X^T X + \beta I_p\right)^{-1} X^T Y$$

where I_p is the p-dimensional identity matrix. The resulting estimator $\hat{\beta}_\lambda$ is called **shrinkage estimator**.

[1] https://blog.csdn.net/qq_20259459/article/details/70316511

3.3 Kernel Regression

Given training samples $\{(X_i, y_i)\}_{i=1,2,\cdots,n}$, where X_i is a p-dimensional vector and y_i is the ground-truth. The kernel regression wants to learn a regression model of the form

$$y = f_c(X) = \sum_{i=1}^{n} c_i K(X, X_i)$$

where K is the kernel function introduced in chapter 2.

The goal of kernel regression is to minimize the loss function

$$\mathcal{L}(c) = \sum_{i=1}^{n} \left\| y_i - \sum_{j=1}^{n} c_j K(X_i, X_j) \right\|^2 + \lambda \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j K(X_i, X_j)$$

Let K be the $n \times n$ kernel matrix with $K_{ij} = K(X_i, X_j)$, then the objective function in matrix form is

$$\mathcal{L}(c) = \|Y - Kc\|^2 + \lambda c^T K c$$

where $\|Y - Kc\|^2$ is the conventional square loss in kernel version, $\lambda c^T K c$ is the regularization term penalizing the weight for $\phi(X)$ with large absolute values, which prevents the regression from being conducted based on a few feature dimensions and boosts the robustness of the model, and λ is a hyper-parameter of the model.

Thus, we want to find $\hat{c}_\lambda$ with minimal loss.

$$\hat{c}_\lambda = \arg\min_c \mathcal{L}(c) = \arg\min_c \|Y - Kc\|^2 + \lambda c^T K c$$

The first order condition of the optimization problem is

$$\frac{\partial \mathcal{L}(c)}{\partial c} = 2K^T K c - 2K^T Y + 2\lambda K c = 0$$

Hence, the estimate of c is

$$\hat{c}_\lambda = (K^T K + \lambda K)^{-1} K^T Y$$

Notice for kernel matrix, we have $K = K^T$, therefore,

$$\hat{c}_\lambda = (K^T K + \lambda K)^{-1} K^T Y = (K + \lambda I_n)^{-1} K^{-1} K^T Y = (K + \lambda I_n)^{-1} Y$$

where I_n is the n-dimensional identity matrix.

3.4 Spline Regression

Given training samples $\{(x_i, y_i)\}_{i=1,2,\cdots,n}$, where x_i is one-dimensional and y_i is the ground-truth. Suppose we have a set of knots k_j $(j = 1,2,\cdots,p)$, and we fit a linear spline of the form

$$y = f(x) = \alpha_0 + \sum_{j=1}^{p} \alpha_j \max(0, x - k_j)$$

by minimizing the loss function as

$$\mathcal{L}(\alpha) = \sum_{i=1}^{n} \left\| y_i - \alpha_0 - \sum_{j=1}^{p} \alpha_j \max(0, x - k_j) \right\|^2 + \lambda \sum_{j=1}^{p} \alpha_j^2$$

where $\alpha = [\alpha_0, \alpha_1, \cdots, \alpha_p]^T$ is the parameter vector of the model.

Let $\tilde{X}$ be an $n \times p$ matrix where $\tilde{X}_{ij} = \max(0, x_i - k_j)$. Let $Z = [\mathbf{1}_n \quad \tilde{X}]$ be an $n \times (p+1)$ matrix and $D = diag(0,1,1,\cdots,1)$ be a $(p+1) \times (p+1)$ matrix. Therefore, the objective function can be written as

$$\mathcal{L}(\alpha) = \|Y - Z\alpha\|^2 + \lambda \|D\alpha\|^2$$

Thus, we want to find $\hat{\alpha}_\lambda$ with minimal loss.

$$\hat{\alpha}_\lambda = \arg\min_\alpha \mathcal{L}(\alpha) = \arg\min_\alpha \|Y - Z\alpha\|^2 + \lambda \|D\alpha\|^2$$

The first order condition of the optimization problem is

$$\frac{\partial \mathcal{L}(\alpha)}{\partial \alpha} = 2Z^T Z\alpha - 2Z^T Y + 2\lambda D\alpha = 0$$

Notice that here we use the property $D = D^T D$ to simplify the equation.

Hence, the estimate of α is

$$\hat{\alpha}_\lambda = (Z^T Z + \lambda D)^{-1} Z^T Y$$

3.5 Relationship among Ridge Regression, Kernel Regression and Spline Regression

Let we list the loss function of three regressions as follows.

$$\mathcal{L}(\beta) = \|Y - X\beta\|^2 + \lambda\|\beta\|^2$$
$$\mathcal{L}(c) = \|Y - Kc\|^2 + \lambda c^T K c$$
$$\mathcal{L}(\alpha) = \|Y - Z\alpha\|^2 + \lambda\|D\alpha\|^2$$

We can find out that they all have a similar form, that is, the square loss plus a regularization term.

Ridge Regression & Kernel Regression.

First, we discuss the relationship between ridge regression and the kernel regression. Let us rewrite the regularization term of the kernel regression into this form:

$$c^T K c = \sum_{i=1}^{n}\sum_{j=1}^{n} c_i K(X_i, X_j) c_j = \sum_{i=1}^{n}\sum_{j=1}^{n} c_i \phi(X_i)^T \phi(X_j) c_j = \left(\sum_{i=1}^{n} c_i \phi(X_i)\right)^T \left(\sum_{i=1}^{n} c_i \phi(X_i)\right) \overset{\text{def}}{=} \beta^T \beta = \|\beta\|^2$$

Therefore, if we interpret β as the model parameters, the kernel regression has the form of ridge regression.

Ridge Regression & Spline Regression.

Then, we discuss the relationship between ridge regression and the spline regression. If we regard $D\alpha$ as the model parameters in the spline regression, the spline regression also has the form of ridge regression.

Kernel Regression & Spline Regression.

Finally, let us focus on the relationship between kernel regression and the spline regression. If we let the knots of spline regression be $x_1, x_2, \cdots, x_n$, and we let "kernel function" $K(x_i, x_j)$ denote $\max(0, x_i - x_j)$. Hence, the spline regression model coincides with a kernel regression, *i.e.*,

$$y = f(x) = \sum_{j=1}^{n} \alpha_j K(x, x_j)$$

with $\alpha_0 = 0$ in the spline regression.

Notice that we use quotation mark in the phase "kernel regression", since $K(x_i, x_j)$ actually is not a kernel. Although K satisfies the property of positive semi-definite, K does not satisfy the symmetry of kernel matrix. Hence, we just regard K as a virtual "kernel", and we can find out the relationship between the spline regression and the kernel regression.

3.6 Lasso Regression

Given training samples $\{(X_i, y_i)\}_{i=1,2,\cdots,n}$, where X_i is a p-dimensional vector and y_i is the ground-truth. The Lasso regression is also based on a linear regression model $y = X\beta + \varepsilon$, where X is an $n \times p$ matrix consisting of X_i, Y is an $n \times 1$ vector of ground truth, and $\varepsilon \sim N(0, \sigma^2 I_n)$. Then, similar to the ridge regression except make substitution to the regularization term, Lasso regression defines the loss function as follows.

$$\mathcal{L}(\beta) = \frac{1}{2}\|Y - X\beta\|^2 + \lambda\|\beta\|_{\ell_1}$$

where $\|\beta\|_{\ell_1} = \sum_{j=1}^{p}|\beta_j|$ is the L1 norm introduced in Chapter 2.9. Thus, we want to find $\hat{\beta}_\lambda$ with minimal loss.

$$\hat{\beta}_\lambda = \arg\min_\beta \mathcal{L}(\beta) = \arg\min_\beta \left[\frac{1}{2}\|Y - X\beta\|^2 + \lambda\|\beta\|_{\ell_1}\right]$$

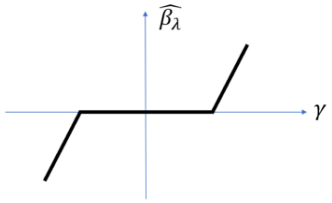
Lasso stands for "least absolute shrinkage and selection operator." Unfortunately, there is no closed form for general p. However, when $p = 1$, *i.e.*, X is an $n \times 1$ vector for n samples, and each sample X_i is a scalar, we do have closed form solution, that is,

$$\hat{\beta}_\lambda = \begin{cases} (\langle Y, X\rangle - \lambda)/\|X\|^2, & \langle Y, X\rangle > \lambda \\ (\langle Y, X\rangle + \lambda)/\|X\|^2, & \langle Y, X\rangle < \lambda \\ 0, & |\langle Y, X\rangle| \le \lambda \end{cases}$$

and we can re-write the solution into the following form.

$$\hat{\beta}_\lambda = \text{sign}(\hat{\gamma}) \max\left(0, |\hat{\gamma}| - \frac{\lambda}{\|X\|^2}\right)$$

where $\hat{\gamma} = \langle Y, X\rangle/\|X\|^2$ is the least square estimator, and the above transformation from $\hat{\gamma}$ to $\hat{\beta}_\lambda$ is called soft thresholding, whose figure is illustrated on the right.



Comparison between Lasso regression and ridge regression.

$$\hat{\beta}_\lambda^{Lasso} = \text{sign}(\hat{\gamma}) \max\left(0, |\hat{\gamma}| - \frac{\lambda}{\|X\|^2}\right), \qquad \hat{\beta}_\lambda^{ridge} = \frac{\langle Y, X\rangle}{\|X\|^2 + \lambda}$$

The behavior of Lasso regression is richer than the behavior of ridge regression, including shrinkage (by subtracting λ) and selection (by thresholding at λ); while the ridge regression only includes shrinkage (by adding λ in the denominator).

- Ridge regression implies no dominating features, since it penalizes elements with large absolute values in β, *i.e.*, avoiding using very few feature dimensions for regression. The model uses a large number of feature dimensions for regression, and each dimension contributes a little to the regression result.
- Lasso regression implies sparse features, since it prefers sparse β, *i.e.*, only a small number of components of β are non-zero.

Primal Form of Lasso Regression.

The primal form of Lasso regression is:

$$\min_{\beta} \quad \frac{1}{2}\|Y - X\beta\|^2$$
$$\text{s.t.} \quad \|\beta\|_{l_1} \le t$$

and the dual form of Lasso regression is:

$$\min_{\beta} \frac{1}{2}\|Y - X\beta\|^2 + \lambda\|\beta\|_{\ell_1}$$

which can be derived from the primal form using methods like Lagrange multipliers. The two form is equivalent with a one-to-one correspondence between t and λ, *i.e.*, if $\hat{\beta}_\lambda$ is the solution to the dual form, then it must be the solution to the primal form with $t = \|\hat{\beta}_\lambda\|_{\ell_1}$. The specific proof of the property is given as follows.

Proof of the one-to-one correspondence property between λ and t. (by contradiction)

Suppose there exist a better solution $\widehat{\beta}$ to the primal form, which is different from the solution $\hat{\beta}_\lambda$ to the dual form. Then,

$$\frac{1}{2}\|Y - X\hat{\beta}\|^2 < \frac{1}{2}\|Y - X\hat{\beta}_\lambda\|^2$$

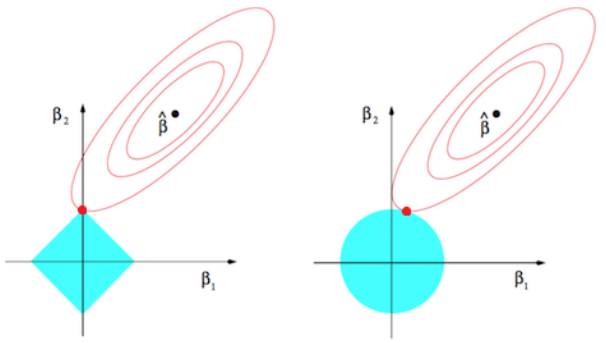
Since $\hat{\beta}$ satisfies the condition that $\|\hat{\beta}\|_{\ell_1} \le t = \|\hat{\beta}_\lambda\|_{\ell_1}$, thus

$$\frac{1}{2}\|Y - X\hat{\beta}\|^2 + \lambda\|\hat{\beta}\|_{\ell_1} < \frac{1}{2}\|Y - X\hat{\beta}_\lambda\|^2 + \|\hat{\beta}_\lambda\|_{\ell_1}$$

Therefore, $\hat{\beta}$ should be the solution to the dual form instead of $\hat{\beta}_\lambda$, which contradicts the premise.

Q.E.D.

The primal form also reveals the sparsity inducing property of ℓ_1 regularization since that ℓ_1 ball has low-dimensional corners, edges, and faces, but is still barely convex, which is illustrated in the right figures. The red circle has the same value of $\|Y - X\beta\|^2$, *i.e.*, the same initial loss term. The solution to the primal form is where the red circle touches the blue region. The reason that ℓ_1 regularization induces sparsity is that it is likely for the red circle to touch the blue region at a corner, which is usually a sparse solution. But if we use ℓ_2 regularization, the solution is not sparse in general.



Sparse, only use β_2 not use β_1 Not sparse use both β_1 and β_2

3.7 Coordinate Descent for Lasso Regression Path

In the previous section, we only provide the close form solutions of single-dimensional feature. In this section, we can use the coordinate descent algorithm to compute $\hat{\beta}_\lambda$, which is based on the idea that we update one parameter at a time, using the close form solutions of single-dimensional features since we only update one parameter. Repeat the process until the parameters converge.

More formally, the algorithm updates one component of parameters at a time, *i.e.*, given the current value of $\beta = \{\beta_j\}_{j=1,2,\cdots,p}$, let $R_j = Y - \sum_{k\neq j} X_k\beta_k$ be the residual when β_k are fixed for all $k \neq j$, then we can update

$$\beta_j = \text{sign}(\hat{\gamma}_j)\max\left(0, |\hat{\gamma}_j| - \frac{\lambda}{\|X_j\|^2}\right)$$

where $\hat{\gamma}_j = \langle R_j, X_j\rangle / \|X_j\|^2$.

The step can be regarded as regress the residual R_j on one-dimensional feature X_j. By repeating the process, more parts of the global residual R can be explained by the regression, and the process converges when all explainable part of global residual R is explained by the parameters β.

The coordinate descent algorithm finds the solution path of Lasso by starting from a relatively big λ and all components of the parameter setting to 0. Then, we gradually reduce λ, and for each λ we cycle through all component of the parameter for coordinate descent until convergence, and then we lower λ. This gives $\hat{\beta}(\lambda)$ for whole range of λ. The whole process is a forward selection process, which sequentially selects new variables and occasionally removes selected variables.

The pseudo-code of the coordinate descent algorithm is shown as follows.

for $\lambda = 10^a, 10^{a-\Delta}, 10^{a-2\Delta}, 10^{a-3\Delta}, \ldots, 10^b$ **do**
  for *Feature dimension* $j = 1, 2, \ldots, p$ **do**
    Compute the residual, $\mathbf{R}_j = \mathbf{Y} - \sum_{k\neq j} \mathbf{X}_k\beta_k$;
    Update the parameter of the j-th dimension, $\beta_j = \text{sign}(\hat{\gamma}_j)\max(0, |\hat{\gamma}_j| - \lambda/\|\mathbf{X}_j\|_{\ell_2}^2)$, where $\hat{\gamma}_j = \langle \mathbf{R}_j, \mathbf{X}_j\rangle / \|\mathbf{X}_j\|_{\ell_2}^2$
  end
end

3.8 Bayesian Regression

Let us interpret the regularization from Bayesian's view. For example, the ridge regression has a Bayesian interpretation. Let $\beta \sim N(0, \tau^2 I_p)$ be the prior distribution of β. Then, according to Bayesian Theorem,

$$\Pr(\beta \mid X, Y) = \frac{\Pr(\beta)\Pr(Y \mid X, \beta)}{\Pr(Y \mid X)}$$

Since $\Pr(Y \mid X)$ is a constant value determined by the problem model when X, Y are given[2], we can write the log probability density of β when given X, Y as follows.

$$\log\Pr(\beta \mid X, Y) = \log\Pr(\beta) + \log\Pr(Y \mid X, \beta) + C$$

The assumption of the least square models shows that $Y \mid X, \beta \sim N(X\beta, \sigma^2 I)$ [3], therefore

$$\log\Pr(\beta \mid X, Y) = -\frac{1}{2\tau^2}\|\beta\|^2 - \frac{1}{2\sigma^2}\|Y - X\beta\|^2 + C$$

Therefore, according to the maximum a posteriori (MAP) estimation, we want to maximize the previous formula. Therefore, the solution to the optimization problem is as follows.

$$\hat{\beta} = \left(X^T X + \frac{\sigma^2}{\tau^2} I_p\right)^{-1} X^T Y$$

which corresponds to the ridge regression with $\lambda = \sigma^2/\tau^2$. Hence, the ridge regression has an interpretation from Bayesian's view, that is, the ridge regression uses the knowledge of the prior distribution of β to find out the optimal parameters for the given problem.

3.9 General Ridge Regression: Linear Version and Feature Version

Given training samples $\{(X_i, y_i)\}_{i=1,2,\cdots,n}$, where X_i is a p-dimensional vector and y_i is the ground-truth. Consider the general ridge loss of the linear model shown as follows.

$$\mathcal{L}(\beta) = \sum_{i=1}^{n} L(y_i, X_i^T\beta) + \lambda\|\beta\|^2$$

[2] https://www.mit.edu/~9.520/spring09/Classes/class15-bayes.pdf
[3] https://statisticaloddsandends.wordpress.com/2018/12/29/bayesian-interpretation-of-ridge-regression/

where $L(y_i, \hat{y}_i)$ is the sample loss function between the ground truth y_i and the prediction $\hat{y}_i$. Notice that here we use the formal form $X_i^T\beta$ to represent the prediction instead of the convenient form $X_i\beta$ used in earlier sections. We will show that the minimizer can be written in the form of

$$\hat{\beta} = \sum_{j=1}^{n} \alpha_j X_j$$

Proof. (by contradiction)

Suppose we need additional vectors, orthogonal to the X_j's ($j \in \{1,2,\cdots,n\}$), to generate the solution to the problem. Call that alternative solution $\tilde{\beta}$. Without loss of generality, assume that we need K additional perpendicular vectors to generate $\tilde{\beta}$, where K can be any positive integer different from 0. Then, we can write $\tilde{\beta}$ as $\tilde{\beta} = \sum_{j=1}^{n} \alpha_j X_j + \sum_{k=1}^{K} \kappa_k X_k'$, with $X_k' \perp X_i$ for all i and k.

Notice that

$$X_i^T\tilde{\beta} = X_i^T\left(\sum_{i=1}^{n} \alpha_j X_j + \sum_{k=1}^{K} \kappa_k X_k'\right) = \sum_{i=1}^{n} \alpha_j X_i^T X_j + \sum_{k=1}^{K} \kappa_k X_i^T X_k' = \sum_{i=1}^{n} \alpha_j X_i^T X_j = X_i^T\hat{\beta}$$

since $X_k' \perp X_i$, which means $\langle X_i, X_k' \rangle = X_i^T X_k' = 0$. Therefore, the sample loss function $L(y_i, X_i^T\beta)$ under $\tilde{\beta}$ take on the same values than under $\hat{\beta}$, that is, $L(y_i, X_i^T\tilde{\beta}) = L(y_i, X_i^T\hat{\beta})$.

Then, for the regularization term, let $\gamma = \sum_{k=1}^{K} \kappa_k X_k'$, we have

$$\|\tilde{\beta}\|^2 = \|\hat{\beta}\|^2 + \|\gamma\|^2 + 2 \cdot \hat{\beta} \cdot \hat{\gamma} = \|\hat{\beta}\|^2 + \|\gamma\|^2 \geq \|\hat{\beta}\|^2$$

and only when $\|\gamma\|^2 = 0$ the equality holds. Since K is different from 0, we must have $\|\gamma\|^2 > 0$, therefore

$$L(\tilde{\beta}) = \sum_{i=1}^{n} L(y_i, X_i^T\tilde{\beta}) + \lambda\|\tilde{\beta}\|^2 = \sum_{i=1}^{n} L(y_i, X_i^T\hat{\beta}) + \|\hat{\beta}\|^2 + \|\gamma\|^2 > \sum_{i=1}^{n} L(y_i, X_i^T\hat{\beta}) + \|\hat{\beta}\|^2 = L(\hat{\beta})$$

Therefore, $\tilde{\beta}$ cannot be the minimizer, which contradicts the premise.

This property implies that vectors perpendicular to our observations are not helpful in deriving the solution. Note the interesting case where X is $p \times n$, where $p > n$. One could conjecture that n observations in this case are not enough to derive the solution, and we would need p observations instead. We can look at this conjecture by letting $K = p - n$ in the proof above, to see that the n observations are already enough.

Consider the feature version of the property, whose ridge loss is shown as follows.

$$\mathcal{L}(\beta) = \sum_{i=1}^{n} L(y_i, \phi(X_i)^T\beta) + \lambda\|\beta\|^2$$

Then, the minimizer can be written in the form of

$$\hat{\beta} = \sum_{j=1}^{n} \alpha_j \phi(X_j)$$

The proof is analogous to the one above, except that here our features are not X_i, but vectors $\phi(X_i)$.

3.10 Gaussian Process

Mathematical Foundation: Variance.

When random variable A is independent on another random variable B, then the variance of $A + B$ should be the summation of variance of A and variance of B.

$$\begin{aligned} Var[A+B] &= E\left[(A+B-E[A]+E[B])^2\right] \\ &= E\left[(A-E[A])^2\right] + E\left[(B-E[B])^2\right] + 2E[(A-E[A])(B-E[B])] \\ &= E\left[(A-E[A])^2\right] + E\left[(B-E[B])^2\right] + 2E[A-E[A]]E[B-E[B]] \\ &= Var[A] + Var[B] \end{aligned}$$

The derivation uses the following property: $E[A - E(A)] = 0$ for all variable A. And we use the condition that A is independent on B when factoring $E\big[\big(A - E(A)\big)\big(B - E(B)\big)\big]$ into $E[A - E(A)]$ and $E[B - E(B)]$. Hence, for independent random variables, the variance of the summation is equal to the summation of variance.

Mathematical Foundation: Multivariate Normal Distribution.

Notice that if arbitrary random vectors X_1 and X_2 are multivariate normal with joint distribution given by

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right) \stackrel{def}{=} N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \Sigma\right)$$

Then, it must follow that[4]

$$\Pr(X_2 \mid X_1) \sim N(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(X_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12})$$

Linear Version.

Suppose $Y = X\beta + \varepsilon$, where $\beta \sim N(0, \tau^2 I_p), \varepsilon \sim N(0, \sigma^2 I_n)$, and ε is independent of β, X is a row vector.

What we want to find out is the posterior distribution $\Pr(\beta \mid Y, X)$. In the following discussions, we will assume that X is the given condition, so we omit X in the formula to facilitate notation. Noteworthily, we view Y as a random variable following a certain distribution rather than a single observed value, that is, we can interpret the given ground truth Y as an observation to the distribution of Y.

Hence,

$$E[Y] = XE[\beta] + E[\varepsilon] = 0$$
$$Var[Y] = Var[X\beta] + Var[\varepsilon] = X\tau^2 I_p X^T + \sigma^2 I_n = \tau^2 XX^T + \sigma^2 I_n$$

Therefore, $Y \mid X \sim N(0, \tau^2 XX^T + \sigma^2 I_n)$ since both β and ε follow the Gaussian distribution. If we consider the joint distribution of multivariate normal Y and β, we can use the conditional probability formula introduced previously to get $\Pr(\beta \mid Y, X)$ Then, we need to calculate the covariance matrix between Y and β first.

$$\begin{aligned} Cov(Y, \beta) &= E[(Y - E[Y])(\beta - E[\beta])^T] \\ &= E[(X\beta + \varepsilon - E[X\beta + \varepsilon])(\beta - E[\beta])^T] \\ &= E[(X\beta + \varepsilon - E[X\beta] - E[\varepsilon])\beta^T] \\ &= E[(X\beta + \varepsilon)\beta^T] \\ &= E[X\beta\beta^T] \\ &= XVar[\beta] \\ &= \tau^2 X \end{aligned}$$

Notice that in the derivation, we use the property $E[\beta] = 0, E[\varepsilon] = 0$ and β is independent on ε. Similarly, we can derive that $Cov(\beta, Y) = \tau^2 X^T$. Therefore,

$$\begin{bmatrix} Y \\ \beta \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \tau^2 XX^T + \sigma^2 I_n & \tau^2 X \\ \tau^2 X^T & \tau^2 I_p \end{bmatrix}\right)$$

which implies that

$$\beta \mid Y, X \sim N\left(\tau^2 X^T(\tau^2 XX^T + \sigma^2 I_n)^{-1} Y, \tau^2 I_p - \tau^2 X^T(\tau^2 XX^T + \sigma^2 I_n)^{-1}\tau^2 X\right)$$

Therefore, according to the maximum likelihood estimation,

$$\hat{\beta} = \tau^2 X^T(\tau^2 XX^T + \sigma^2 I_n)^{-1} Y = X^T\left(XX^T + \frac{\sigma^2}{\tau^2} I_n\right)^{-1} Y$$

For Gaussian process, we need to add an inference stage, therefore, given a test input X',

$$\hat{y}' = X'\hat{\beta} \sim N\left(X'\tau^2 X^T(\tau^2 XX^T + \sigma^2 I_n)^{-1} Y, XX^T\left(\tau^2 I_p - \tau^2 X^T(\tau^2 XX^T + \sigma^2 I_n)^{-1}\tau^2 X\right)\right)$$

which is the idea of Gaussian process – model the distribution of the prediction based on training samples X, Y.

Bayesian Interpretation of Linear Version.

From Bayesian's view introduced in Chapter 3.8, we know that another approach to reach $\Pr(\beta \mid Y, X)$ is to use the Bayesian Theorem, which gives us the following log-likelihood.

$$\log \Pr(\beta \mid X, Y) = -\frac{1}{2\tau^2}\|\beta\|^2 - \frac{1}{2\sigma^2}\|Y - X\beta\|^2 + C$$

---

[4] Eaton, Morris L. (1983). Multivariate Statistics: a Vector Space Approach. John Wiley and Sons. pp. 116–117.

which is equivalent with ridge regression with $\lambda = \sigma^2/\tau^2$, and the solution is

$$\hat{\beta}' = \left(X^T X + \frac{\sigma^2}{\tau^2} I_p\right)^{-1} X^T Y$$

Actually, the $\hat{\beta}'$ solved from the maximum log-likelihood estimation in the Bayesian's view is equal with the $\hat{\beta}$ derived previously in the linear version, *i.e.*, $\hat{\beta} = \hat{\beta}'$. This is because

$$\left(X^T X + \frac{\sigma^2}{\tau^2} I_p\right) X^T = X^T \left(X X^T + \frac{\sigma^2}{\tau^2} I_n\right) \Longrightarrow \hat{\beta} = X^T \left(X X^T + \frac{\sigma^2}{\tau^2} I_n\right)^{-1} Y = \left(X^T X + \frac{\sigma^2}{\tau^2} I_p\right)^{-1} X^T Y = \hat{\beta}'$$

Hence, the linear version of gaussian process has the same solution as ridge regression.

Feature Version and Kernel Version.

Suppose $y_i = \phi(X_i)^T \beta + \varepsilon_i$, with the same prior as above. Let us look for the distribution of $\beta \mid Y, X$. Let $f(X_i) = \phi(X_i)^T \beta$, and define an $n \times d$ matrix $\phi(X)^T = [\phi(X_1)^T, \phi(X_2)^T, \cdots, \phi(X_n)^T]$ based on the assumption that ϕ is d-dimensional. With the defined $\phi(X)^T$, we have $Y = \phi(X)^T \beta + \varepsilon$. Similar with the feature version we have learnt in Chapter 2, we can simply let $\phi^T(X)$ represent our new set of features, and follow the same steps introduced previously. Therefore, we get

$$\beta \mid Y, X \sim N\left(\tau^2 \phi(X)(\tau^2 \phi(X)^T \phi(X) + \sigma^2 I_n)^{-1} Y, \tau^2 I_p - \tau^2 \phi(X)(\tau^2 \phi(X)^T \phi(X) + \sigma^2 I_n)^{-1} \tau^2 \phi(X)^T\right)$$

For inference stage of Gaussian process, given a test input X',

$$\hat{y}' = \phi(X')^T \hat{\beta} \sim N\left(\tau^2 K(X', X)(\tau^2 K + \sigma^2 I_n)^{-1} Y, \tau^2 K(X', X') - \tau^2 K(X', X)(\tau^2 K + \sigma^2 I_n)^{-1} \tau^2 K(X, X')\right)$$

Notice here we use K without parameters to represent the kernel matrix for the training samples, *i.e.*, an $n \times n$ matrix $K_{ij} = K(X_i, X_j) = \phi(X_i)^T \phi(X_j)$. We abuse notation in that $K(X', X)$ where X' is a testing sample and X is the training samples as an $1 \times n$ vector. Similarly, $K(X, X') = K(X', X)^T$ denotes an $n \times 1$ vector.

Sometimes, we absorb τ^2 into the definition of K, and we can get the following form:

$$\hat{y}' = \phi(X')^T \hat{\beta} \sim N\left(K(X', X)(K + \sigma^2 I_n)^{-1} Y, K(X', X') - K(X', X)(K + \sigma^2 I_n)^{-1} K(X, X')\right)$$

There is a connection with kernel regression. For kernel regression, we know that $\hat{c} = (K + \sigma^2 I_n)^{-1} Y$, so the estimated coefficients in the kernel regression show up in the mean of $\Pr[f(X') \mid Y, X]$, which means that $K(X', X)\hat{c}$ is actually the prediction in inference stage we would get if we did a kernel regression. The variance of the posterior shown above actually allows us to deal with uncertainty, creating a posterior interval for our kernel regression estimate. Therefore, the kernel version allows us to conduct kernel regressions and create posterior intervals for estimates by adding variances to them.

Marginal Likelihood Estimation for Kernel Parameters.

The marginal distribution of Y is $N(0, K_\gamma + \sigma^2 I_n)$, where K is the kernel function with parameter γ. Then, the marginal likelihood aims to determine parameter γ of the kernel. Take Gaussian kernel (RBF kernel) for example, let us write down the marginal likelihood and log marginal likelihood as follows.

$$\mathfrak{L}(\gamma) = \frac{1}{(2\pi)^{n/2} \left|\Sigma_\gamma\right|^{1/2}} \exp\left(-\frac{1}{2} Y^T \Sigma_\gamma^{-1} Y\right)$$

where $\Sigma_\gamma = K_\gamma + \sigma^2 I_n$. Notice that the likelihood is based on the given training samples X and their labels (observations) Y. Then, the log-marginal-likelihood for determining γ is then:

$$\log \mathfrak{L}(\gamma) = -\frac{1}{2} Y^T \Sigma_\gamma^{-1} Y - \frac{1}{2} \log\left|\Sigma_\gamma\right| - \frac{n}{2} \log 2\pi$$

By optimizing the objective, we can get the optimal parameter γ for the given kernel, which is called log marginal likelihood estimation for kernel parameters.