머신러닝 깊게 이해하기

노트테이커: 김성원

주차: 1주차

일시: 6th, October

1 소개

훈련 집합: $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$

표적 벡터: $\mathbf{x}_n$에 대응되는 정답 $t_n \in \mathbf{t}$

즉, 머신러닝은 훈련 집합에 대한 표적 벡터를 예측하는 함수 $\boxed{\mathbf{y}(\mathbf{x})}$ 를 만드는 것이다. 훈련 단계에서 $\mathbf{y}(\mathbf{x})$의 정확한 개형을 결정 짓고, 훈련 집합에서 사용되지 않은 시험 집합을 통해 새로운 예측값을 찾는 일반화를 거친다.

1.1 다항식 곡선 피팅

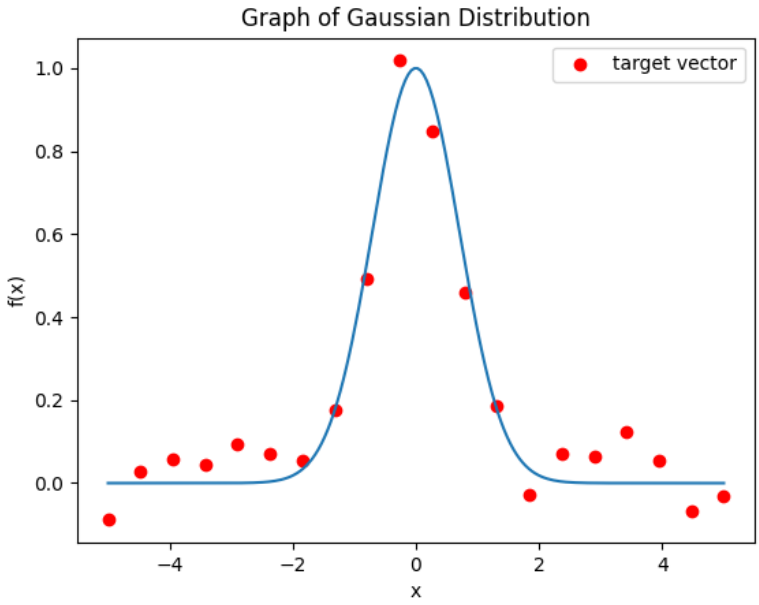

Figure 1: 가우시안 분포와 표적벡터

Figure 1으로부터 알 수 있듯, 표적벡터가 반드시 정확한 값이라는 보장은 없다. 해당 표적벡터는 정규분포(loc=0.0, scale=0.05)의 noise를 가지고 있다.

해당 함수는 정규 분포로, $e^{-x^2} = \sum_{n=0}^{\infty} \frac{(-x^2)^n}{n!} = 1 - x^2 + \frac{1}{2}x^4 - \frac{1}{6}x^6 + \cdots$으로 알려져 있다.
이 초월함수를 다항식으로 나타내기 위해서는 다항식의 차수와 다항식의 계수를 정하여야 한다. 즉, 아래와 같이 나타낼 수 있다.

$$y(x, \mathbf{w}) = w_0 + w_1 x^1 + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

이때, 계수가 선형이라는 것에 주목하자. 이러한 모델을 선형모델이라 한다. 머신러닝에서는 임의의 w값으로부터, 미분을 통해 오차를 줄이는 방향을 구하고 오차를 최소로 하는 값을 구한다. 따라서, 여러 가지 오차 함수가 있으며 여기서는 아래 식과 같은 오차제곱합을 이용한다.

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

1.1.1 왜 미분을 해야 하는가

전미분에 의하여 n개의 변수에 대해 $df = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} dx_i$ 이다. 즉, $d\overrightarrow{r} = (dx_1, dx_2, \cdots, dx_n)$ 만큼의 변화에 대해 $df = \nabla f \cdot d\overrightarrow{r}$이다.
만약, ∇f와 $d\overrightarrow{r}$의 방향이 나란하다면 df가 극대화된다. 따라서, $df = 0$이 되는 지점, 즉 $\nabla f = 0$으로 빠르게 가는 방향임을 확보하여야 한다.

테일러 근사에 의하여 $\nabla f(\mathbf{x_0}) \approx 0$인 지점 근처에서 다음과 같다.

$$f(\mathbf{x}_0 + \Delta\mathbf{x}) \approx f(\mathbf{x}_0) + \frac{1}{2!}\sum_i \sum_j \frac{\partial^2 f}{\partial x_i \partial x_j}\Delta x_i \Delta x_j \tag{1}$$

$$= f(\mathbf{x}_0) + \frac{1}{2}\Delta\mathbf{x}^T \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{pmatrix} \Delta\mathbf{x} = f(\mathbf{x}_0) + \frac{1}{2}\Delta\mathbf{x}^T H \mathbf{x} \tag{2}$$

이때, 행렬 H는 f가 연속일 때 대칭행렬이므로 $Q^T = Q^{-1}$인 $H = Q^T \Lambda Q$의 형태로 대각화할 수 있다. 따라서, H의 고유값을 구하여, 그 고윳값 λ_i의 양음부호를 판정하여야 한다. 왜냐하면, $\Delta\mathbf{x}^T H\mathbf{x} = \Delta\mathbf{x}^T H\mathbf{x} = (Q\Delta\mathbf{x})^T \Lambda (Q\Delta\mathbf{x}) = \mathbf{y}^T\Lambda\mathbf{y} = \sum_i \lambda_i y_i^2$이므로, 변화량의 부호를 온전히 λ가 결정하기 때문이다. 극솟값 주변의 점은 항상 극솟값보다 크므로, H의 모든 고윳값은 양수가 되며 극댓값 주변의 점은 항상 극댓값보다 작으므로, H의 모든 고윳값은 음수가 된다.

테일러 근사에 의하여 $f(\mathbf{x}) \approx \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T H(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)$이고 H가 대칭행렬이므로, 양변을 $\mathbf{x}$로 미분하면 $\nabla f \approx H(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)$이다. 따라서, $(\mathbf{x} - \mathbf{x_0})^T \nabla f = (\mathbf{x} - \mathbf{x_0})^T H(\mathbf{x}_0)(\mathbf{x} - \mathbf{x_0})$로 둔다. $\mathbf{x} - \mathbf{x}_0$는 극값으로의 최적의 방향을 나타낸다. 이에 대해 $\nabla f(\mathbf{x})$가 어떠한 상관관계를 가지는지 확인한다.

극솟값 근처에서 항상 H의 고윳값은 0보다 작으므로 내적값은 음수이다. 따라서, $-\nabla f(\mathbf{x})$는 극소점으로의 방향과 같은 방향의 벡터가 된다. 그러나, 극댓값 근처에서는 내적값이 양수이므로, $-\nabla f(\mathbf{x})$는 극대점으로의 방향과 반대 방향의 벡터가 된다. 따라서, 경사하강법 ($\mathbf{w} = \mathbf{w}_0 - \eta\nabla E$) 등의 방법은 극솟값을 찾는 데 유효한 방법이 된다.

해당 논의에서 사용된 H를 헤시안 행렬이라 한다.

본제로 돌아와서, 데이터 포인트가 20개이므로 20차 다항함수를 구성하면 오차제곱합을 0으로 만들 수 있다. 그러나, 그 방식은 과적합 문제를 일으키므로 사용하지 않는다.

이에 대해 구체적으로 알아보자면, 보간 조건 $P(x_i) = y_i$인 데이터 포인트 n개가 존재한다고 가정하고, 이에 맞도록 n차 다항식을 근사한다고 가정하자. 그렇다면, 다음과 같이 나타낼 수 있다.

$$\begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$Vc = y$$

따라서, 계수 행렬인 c는 $c = V^{-1}y$로 나타낼 수 있다. 이때, V를 방데르몽드 행렬이라 한다. 이 행렬의 행렬식은 같은 x_i, x_j가 있을 때, 그때에만 선형 종속이 된다는 사실을 통해 $C \prod_{1 \le i \le j \le n} (x_j - x_i)$ 형태임을 알 수 있다. 그러나, 같은 범위 내에 데이터 포인트들이 많으면 많을수록 곱셈에 의하여 이 값은 매우 작아지게 된다. 따라서, $\det(V^{-1})$가 발산하게 된다. 이는 계수 행렬의 어떤 성분이 매우 커질 수 있음을 의미한다.

따라서, 이러한 문제를 해결하기 위한 기법으로 정규화가 있다. L2 정규화의 수식은 다음과 같다.

$$\tilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

이 수식을 이해하기 위해 라그랑주 승수법을 적용해볼 수 있다. 함수 E와 어떤 제약 조건 $g = 0$에 대하여 $\nabla E = \lambda \nabla g$를 만족하는 점이 제약조건 위에서 E의 극값이 된다. 따라서, 제약 조건 위에서 $\nabla(E + \lambda g) = 0$의 형태가 되는데, E가 오차제곱합이고, $E + \lambda g$를 L2 정규화의 식으로 바라보면, $g = \dfrac{\lambda}{2}\|w\|^2$임을 알 수 있다. 또한, ∇에서 상수항은 무시되므로, $\nabla(E + \lambda(g + \text{Const.}))$는 극점의 위치를 변화시키지 않는다. 결국, 제약 조건이 $\|w\|^2 - \text{Const.} = 0$임을 알 수 있으며, 이 상수값은 λ에 의하여 결정되는 것임을 알 수 있다. 즉, L2 정규화는 가중치의 크기를 어떤 상수로 제한하는 수식임을 알 수 있다.

이를 확인해보기 위해 Fig1과 같은 함수로부터 100개의 훈련 데이터와 100개의 시험 데이터를 구성하고, 10차 다항식으로 피팅 후 $\|\mathbf{w}\|^2$와 시험 데이터를 통해 RMS를 구하여 Table 1에 표현하였다.

Table 1: 10th-order polynomial fitting on Gaussian target ($\sigma = 0.05$ noise)

Coefficient	$\ln\lambda = 0$	$\ln\lambda = -2$	$\ln\lambda = -\infty$
w_0^*	0.833836	0.937539	0.968964
w_1^*	-0.005598	-0.005566	-0.005539
w_2^*	-0.429940	-0.769306	-0.888973
w_3^*	-0.004179	-0.004534	-0.004619
w_4^*	0.000339	0.247967	0.340113
w_5^*	0.002633	0.002822	0.002862
w_6^*	0.031194	-0.038104	-0.064669
w_7^*	-0.000385	-0.000416	-0.000423
w_8^*	-0.005462	0.002788	0.006012
w_9^*	0.000016	0.000017	0.000018
w_{10}^*	0.000274	-0.000077	-0.000216
RMS	0.064000	0.028000	0.022000
$\|\mathbf{w}\|_2^2$	0.881189	1.533817	1.849121

이를 통해, λ가 커질수록 Const.의 값이 작아진다는 사실을 알 수 있으며, 이에 따라 λ가 커질수록 크기가 작은 계수를 취하는 경향을 확인할 수 있다. 다만, 이는 제약 조건 위에서의 극값에 불과하므로 과적합은 방지할 수 있지만, RMS의 감소는 보장하지 못한다.

1.2 가우시안 분포와 곡선 피팅

베이즈 정리는 다음과 같다.

$$p(\mathbf{w}|D) = \frac{p(D|\mathbf{w})p(\mathbf{w})}{p(D)}$$

좌변은 관측값 D가 주어졌을 때, 계수 $\mathbf{w}$에 대한 믿음이며 사후확률이라 한다. 이는, 이미 주어진 결과를 토대로 그것이 어떤 것으로부터 파생되었는지를 파악하는 "예측"에 대한 분포가 된다. 우변의 $p(D|\mathbf{w})$는 $\mathbf{w}$가 주어졌을 때 D가 나타날 확률이며, 이를 가능도 함수라 한다. $p(D)$는 관측값 자체의 믿음이며, 적분값이 1이 되도록 해준다. $p(\mathbf{w})$는 정해지지 않은 계수의

믿음이므로, 어떤 분포로 나타내어지며, 이를 사전 분포라 한다.

결국, 머신러닝은 결괏값만을 입력 받는다는 점을 감안하면, 우리가 얻고자 하는 계수는 가능도함수와 사전확률에 비례하는 분포를 가진다는 것을 알 수 있다.

또한 사후확률 중에서도 가장 믿음이 큰 $\mathbf{w}$를 얻어야 하므로, 가능도 함수가 최대가 되는 경우를 살펴보아야 한다. 가능도를 최대화 시켰을 때의 값을 최대 가능도라고 하며, 이를 유도할 수 있는 $\mathbf{w}$를 선택한다.

모집단의 분포와 관계 없이, 표본집단의 표본 수가 증가함에 따라 표본의 분포는 가우시안 분포로 분포수렴함을 중심극한정리로부터 알 수 있다. 따라서, 이 절에서는 모든 데이터가 가우시안 분포를 가진다고 가정한다.

따라서, 사전 분포와 가능도 함수 모두 가우시안 분포를 따르므로 각각 아래와 같이 표현할 수 있다.

$$p(t|x, \mathbf{w}, \beta) = N(t|y(x, \mathbf{w}), \beta^{-1})$$

이때, 분산에 놓인 β를 정밀도라고 한다.

$$p(\mathbf{w}|\alpha) = N(\mathbf{w}|\mathbf{0}, \alpha^{-1}I) = \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2} \exp\left\{-\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}\right\}$$

일반적으로 $p(\mathbf{w}|\alpha)$의 지수항은 $\mathbf{w}^T\Sigma^{-1}\mathbf{w}$로 나타나는데, 공분산 행렬로 항등행렬을 사용한 이유는 모든 다항식의 계수가 서로 독립적이고 분산이 α^{-1}인 단변량 가우시안 분포로 나타날 것이라는 믿음에 기인한다. 즉, $\mathbf{w}$는 i.i.d이다.

연속확률변수에서도 곱의 법칙이 성립함에 따라, 어떤 독립적이고 동일한 분포를 가진 (i.i.d) x가 주어졌을 때 그것이 표적벡터 $\mathbf{t}$를 생성할 확률은 다음과 같다.

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = \prod_{n=1}^{N} N(t_n|y(x_n, \mathbf{w}), \beta^{-1})$$

최대 가능도를 구하기 위해 식을 미분해야 하지만, 그 형태가 복잡하므로 로그를 이용한다.

$$\ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = -\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w})\}^2 + \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi)$$

해당 식을 $\mathbf{w}$의 관점으로 보았을 때, 가능도 함수를 최대화할 수 있는 방법은

$-\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w})\}^2$를 최소화하는 것이다. 따라서, 가우시안 분포로부터 제곱합 오차가 유도됨을 확인할 수 있다.

해당 식으로부터 적절한 $\mathbf{w}$를 구하고, $\ln\beta$항을 이항하여 β에 대해 편미분하면 가능도 함수를 최대화하는 β 또한 구할 수 있다.

$$\frac{1}{\beta}=\frac{1}{N}\sum_{i=1}^{N}\{y(x_n,\mathbf{w})-t_n\}^2$$

또한, 가능도 이외에 사전확률에 대해서도 사후확률을 최대화하여야 하므로 가능도 함수와 사전확률의 곱의 로그를 최대화하는 것을 고려하자.

이때, 정밀도는 상수이므로 배제하고 실질적으로 조정할 수 있는 항만 남기면 아래와 같다.

$$\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w})-t_n\}^2+\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}$$

즉, L2 정규화에서 λ가 $\frac{\alpha}{\beta}$에 대응되는 것을 확인할 수 있다. 이는, $\mathbf{w}$가 거의 원점에 근처에만 존재하도록 제한할수록(α가 클수록), t가 실제값 근처에 느슨하게 나타남을 허용할수록(β가 낮을수록) 곡선이 유연하고 단순해지는 경향을 보였던 L2 정규화의 직관과 잘 들어맞는다.

반대로 베이지안 방법으로 일반화를 하려면 $\mathbf{x}$와 표적 벡터 $\mathbf{t}$가 주어졌을 때, 새로운 입력값 x에 대해 $p(t|x,\mathbf{x},\mathbf{t})$를 구하여야 한다.

이는 합의 법칙과 곱의 법칙을 이용하여 다음과 같이 나타낼 수 있다.

$$p(t|x,\mathbf{x},\mathbf{t})=\int p(t|x,\mathbf{w})p(\mathbf{w}|\mathbf{x},\mathbf{t})d\mathbf{w}$$

1.3 차원의 저주

저차원의 상식을 고차원에 적용할 때, 많은 문제가 발생할 수 있으므로 항상 조심하여야 한다. 예를 들어, 데이터 분류 모델에서 특정 단위 구획으로 나누는 방식은 저차원에서는 크지 않지만, 고차원으로 갈수록 기하급수적으로 증가한다. 이는 계산량의 폭증과 더불어 성능 저하로

이어진다.

심지어는 모델이 데이터의 특징을 잘 담지 못할 수도 있다. D차원에서 매우 얇은 구껍질의 부피는 $V_D(r) = K_D r^D$로 나타낼 수 있다. 만약 아주 작은 ϵ에 대해 구껍질이 차지하는 비중을 계산하면 $\dfrac{V_D(1) - V_D(1-\epsilon)}{V_D(1)} = 1 - (1-\epsilon)^D$으로, 차원이 높아질수록 그 값이 1에 수렴함을 알 수 있다. 즉, 높은 차원에서 구껍질에 해당하는 데이터를 배제(차원 감소)하게 되면 매우 큰 지분의 데이터 특징을 잃어버릴 수 있다.