



수학 1팀

김영호

다룰 내용:

- 1.1. 선형근사와 테일러 전개, 뉴턴 방법
- 1.2. 편미분과 전미분

중점적으로 다루면 좋은 내용:

1. 근사와 error의 관계
2. 뉴턴 방법의 특성(error를 얼마나 줄여주는가?)
3. 뉴턴 방법의 장단점
4. 편미분에서의 chain rule

1.1. 선형근사와 테일러 전개, 뉴턴방법

(1) 선형근사

여러분들은 고등학교 미적분 시간에 접선의 방정식에 대해 학습하였다. 미분 가능한 함수 $f(a)$ 위의 한 점 $(a, f(a))$ 에서의 접선의 방정식은 다음과 같다.

$$\begin{aligned} f(x) - f(a) &= f'(x)(y - a) \\ f(x) &= f(a) + f'(x)(y - a) \end{aligned}$$

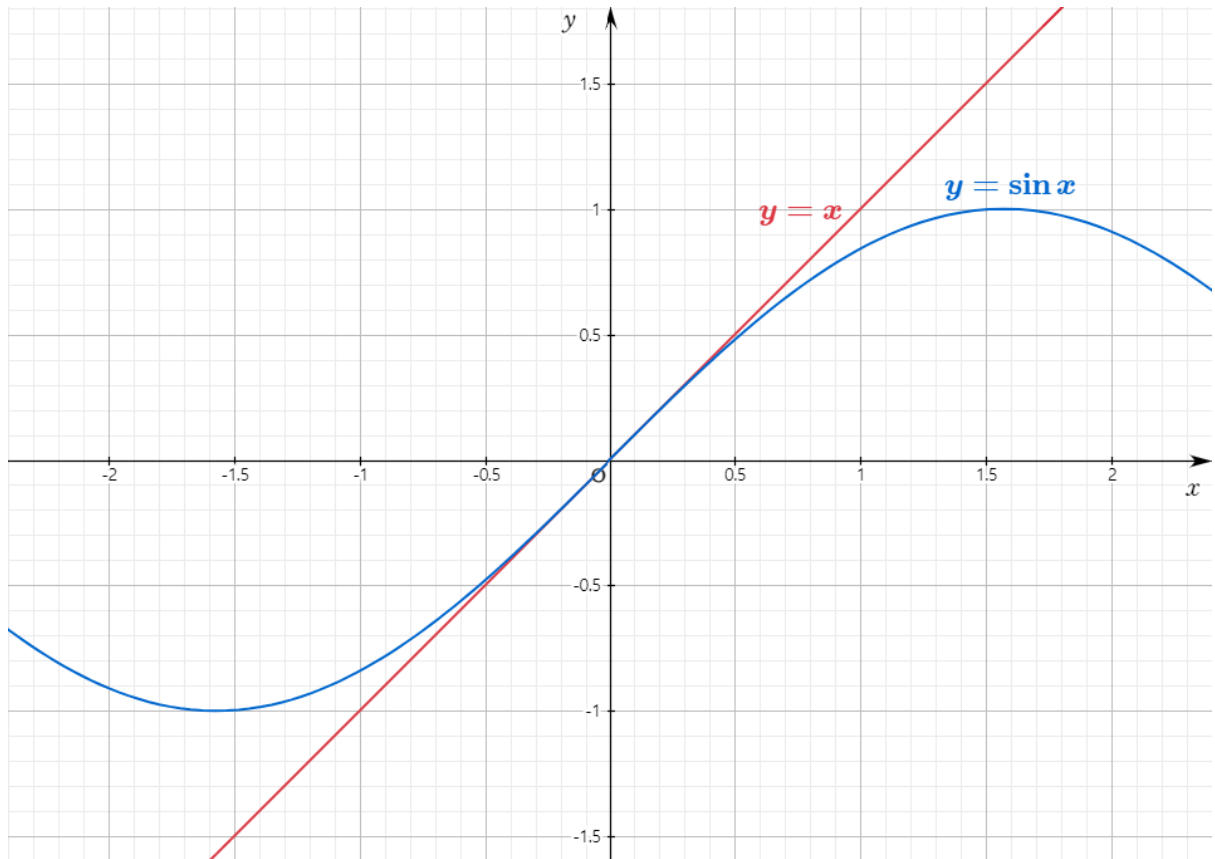
이는 $x = a$ 에서 그은 접선의 방정식의 $x = a$ 근방의 값들이 원함수 $f(x)$ 의 $x = a$ 근방의 값들과 유사함을 의미한다. 이처럼 어느 한 점에서 접선의 방정식을 통해 그 근방에 있는 원함수 함수값의 근사값을 구하는 것을 **선형근사**라고 한다. 그렇다면 다음과 같이 표현할 수 있을 것이다.

$$f(x) \approx f(a) + f'(x)(y - a)$$

또한 $x = a$ 에서의 접선의 함수를 $L(x)$ 로 표기하고, $x = a$ 에서의 $f(x)$ **선형화**라고 한다.

$$L(x) = f(a) + f'(x)(y - a)$$

$\sin(x)$ 함수를 예로 들어 살펴보자. $x = 0$ 에서 그은 접선의 방정식 $y = x$ 의 $x = 0$ 근방의 값들이 $\sin(x)$ 의 $x=0$ 근방의 값들과 거의 차이나지 않음을 다음의 그래프를 통해 알 수 있다.



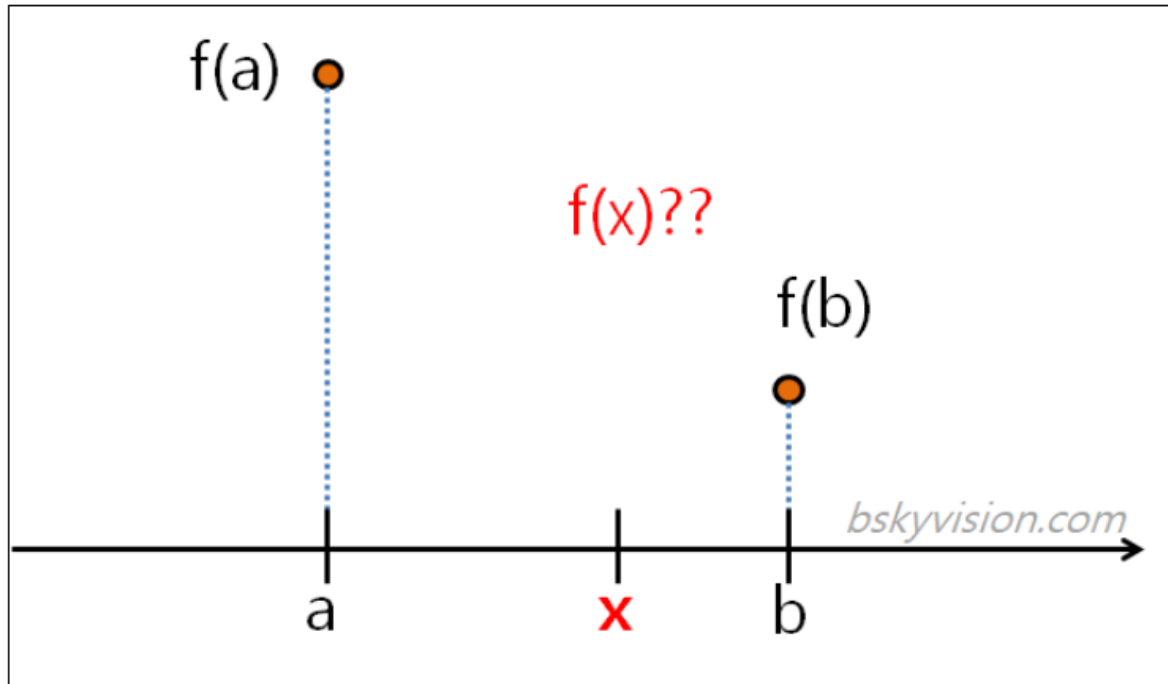
$\sin(x)$ 함수에서 $x = 0$ 일 때 $\sin(0) = 0$ 임은 자명하다. 그러나 $\sin(0.2)$ 의 값을 누군가에게 물어본다면 매우 당황할 것이다. 이러한 경우 선형근사를 이용할 수 있고, $\sin(0.2) \approx 0.2$ 가 된다.

또 다른 예제로 $y = \sqrt{x}$ 의 경우를 살펴보기로 한다. 선형근사를 통해 $\sqrt{4.2}$ 의 값을 구해보면 $\sqrt{4.2} \approx 2.05$ 가 된다. 실제로 $\sqrt{4.2} = 2.0439\dots$ 이므로 상당히 가까운 값을 구했음을 알 수 있다. 여기서 우리는 예측값과 실제값의 차이에 해당하는 오차에 대해 생각해볼 수 있을 것이다. 이 예제의 경우에서 $x = 4$ 를 기준으로 여러 x 값을 통해 계산한 예측값, 실제값, 그리고 오차는 다음과 같다.

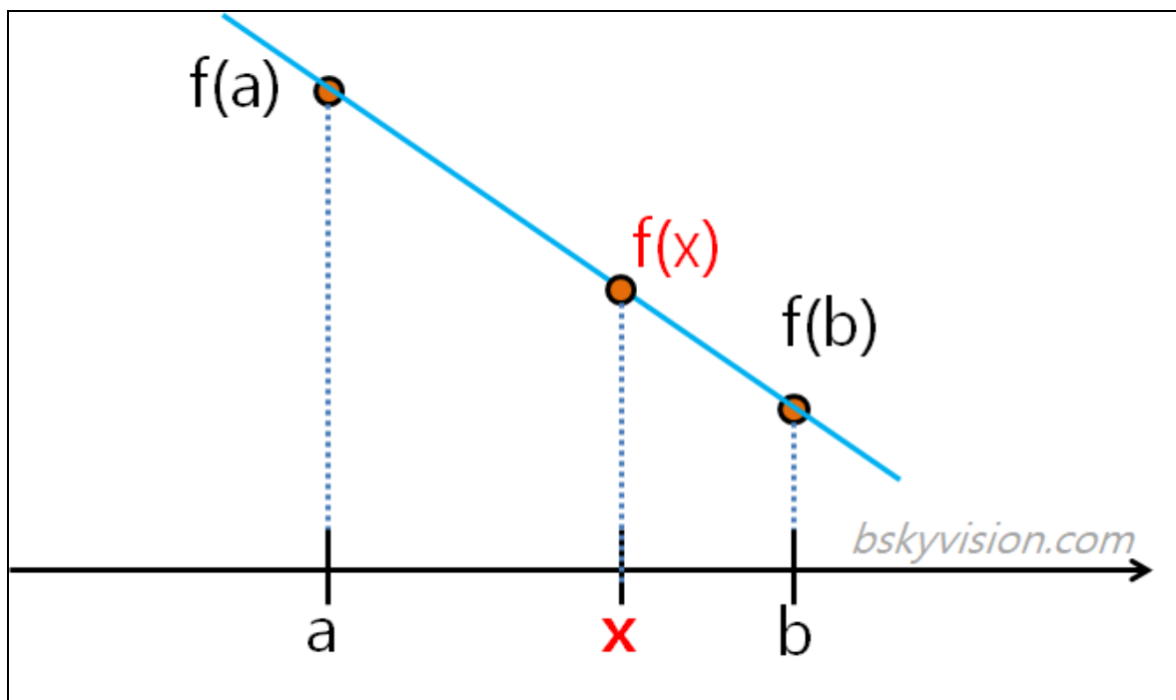
$$\begin{aligned}
 L(3) &= 1.75 & \sqrt{3} &= 1.732\dots & \text{error} &= 0.018 \\
 L(3.5) &= 1.875 & \sqrt{3.5} &= 1.870\dots & \text{error} &= 0.005 \\
 L(4) &= 2 & \sqrt{4} &= 2 & \text{error} &= 0 \\
 L(4.5) &= 2.125 & \sqrt{4.5} &= 2.121\dots & \text{error} &= 0.004 \\
 L(5) &= 2.250 & \sqrt{5} &= 2.236\dots & \text{error} &= 0.014 \\
 L(5.5) &= 2.375 & \sqrt{5.5} &= 2.345\dots & \text{error} &= 0.030
 \end{aligned}$$

이를 통해 $x = 4$ 에서 멀리 떨어질수록 오차값이 커짐을 확인할 수 있다. 따라서 $x = a$ 에서의 선형근사 시 $x = a$ 근방에 있을수록 더욱 정확한 값을 얻어낼 수 있으므로 원하는 오차범위 안에 들도록 적절한 a 의 값을 택해주어야 한다.

추가적으로 **선형보간법(linear interpolation)**에 대해 짚고 넘어가보겠다. 보간법은 지난 학기 세미나를 진행하면서 종종 등장했던 개념이기 때문에 알아두면 유용할 것으로 생각된다. 먼저 보간법이란 두 점이 어떤 값인지 알고 있는지 알고 있는 상황에서 두 점 사이에 있는 어떤 지점의 값을 추정하는 기법을 말한다. 그림으로 나타내보면 다음과 같다.



선형보간법은 위의 그림에서 a 와 b 두 점을 연결한 직선을 이용하여 $f(x)$ 를 추정하는 방법이다. 다음의 그림으로 표현할 수 있을 것이다.



위의 그림으로부터 1차 함수 $y = mx + n$ 상 위에 존재하는 두 점 $(a, f(a)), (b, f(b))$ 를 대입하여 1차 함수 식을 구해보면 다음의 식이 도출된다.

$$y = \frac{f(a) - f(b)}{a - b}x + \frac{af(b) - bf(a)}{a - b}$$

위의 식을 이용하여 a 와 b 사이에 존재하는 점의 함수값 $f(x)$ 를 추정해내는 것이 선형보간법이다. 선형보간법에서는 2개의 점을 이용하여 $f(x)$ 를 추정하는 반면, **삼차보간법(cubic interpolation)**에서는 4개를 통해 $f(x)$ 를 추정한다. 이와 관련된 설명은 reference로 올려놓은 블로그를 살펴보기 바란다.

(2) 테일러 정리

함수 $f(x)$ 가 닫힌 구간 $[a, b]$ 에서 $(n - 1)$ 번 미분 가능하고 열린 구간 (a, b) 에서 n 번 미분 가능하면, 각각의 $x \in [a, b]$ 에 대해,

$$f(x) = f(a) + \frac{f'(a)}{1!}(x - a) + \dots + \frac{f^{(n-1)}(a)}{(n-1)!}(x - a)^{n-1} + \frac{f^{(n)}(\xi)}{n!}(x - a)^n$$

where $a < \xi < x$

이 성립한다.

• 적률생성함수(Mgf; Moment generating function)

테일러 정리가 활용되는 경우로 적률생성함수와 delta-method에 대해 살펴보기로 하자. 먼저 적률생성함수는 X 의 n 제곱의 기댓값 $E(X^n)$ 으로 정의된다. 1차 적률은 X 의 기댓값, 2차 적률은 X 제곱의 기댓값 이런 식이다. 다음의 식에서 양변을 t 에 대해 n 번 미분하고 t 자리에 0을 넣으면 n 차 적률을 구할 수 있다.

$$M_x(t) = E(e^{tx})$$

e^{tx} 은 테일러 정리에서 n 을 무한대로 보내어 끝없이 더하고 a 의 값이 0인 경우인 맥클로린 급수를 이용해 표현 가능하다.

$$e^{tx} = \frac{1}{0!} + \frac{t}{1!}x + \frac{t^2}{2!}x^2 + \frac{t^3}{3!}x^3 + \dots$$

여기서 양변에 기댓값을 취하면, 적률생성함수가 완성된다.

$$M_x(t) = E(e^{tx}) = 1 + tE(x) + \frac{t^2}{2!}E(x^2) + \frac{t^3}{3!}E(x^3) + \dots$$

양변을 t 에 대해 미분한 뒤 t 에 0을 집어 넣으면 1차 적률 기댓값이 된다.

$$\begin{aligned} \frac{dM_x(t)}{dt} &= E(x) + tE(x^2) + \frac{t^2}{2!}E(x^3) + \dots \\ \frac{dM_x(0)}{dt} &= E(x) \end{aligned}$$

한 번 더 t 에 미분한 뒤 $t=0$ 을 집어 넣으면 2차 적률 기댓값이 된다.

$$\begin{aligned}\frac{d^2 M_x(t)}{dt^2} &= E(x^2) + tE(t^3) + \dots \\ \frac{d^2 M_x(0)}{dt^2} &= E(x^2)\end{aligned}$$

이를 일반화하여 t 에 대해 n 번 미분한 뒤에 $t=0$ 을 집어넣으면 n 차 적률을 구하는 다음의 식이 성립된다.

$$\frac{d^n M_x(0)}{dt^n} = E(x^n)$$

그렇다면 실제로 적률생성함수를 어떻게 구할 수 있을까? 확률변수를 X 라 하고, X 의 확률밀도함수를 $f(x)$ 라고 하면, 다음과 같이 적률생성함수를 계산할 수 있다.

$$M_x(t) = \int_{x_1}^{x_2} e^{tx} f(x) dx$$

만약 X 가 정규분포를 따른다면, 다음과 같이 쓸 수 있다.

$$M_x(t) = \int_{-\infty}^{\infty} e^{tx} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx$$

이렇게 적률생성함수를 구하는 이유는 한 번 구해놓으면 적률생성함수를 미분함으로써 X 의 통계량을 쉽게 구할 수 있기 때문이다.

• Delta-method

다음으로 delta-method에 대해 살펴보자. 테일러 정리에서, $f(x)$ 를 1차 미분한 항까지만 고려하면,

$$f(x) \approx f(a) + \frac{f'(a)}{1!}(x-a) \dots (*)$$

가 된다(선형근사식과 같음). 여기서, 평균이 μ 이고 분산이 σ^2 인 확률변수 Y 를 정의하자. 그러면, 우리는 미분 가능한 함수 $f(\cdot)$ 에 대해 $f(Y)$ 의 평균과 분산을 구할 수 있다. (*)식에 의해,

$$f(Y) \approx f(\mu) + \frac{f'(\mu)}{1!}(Y-\mu)$$

가 되고, 평균과 분산을 취해보면,

$$\begin{aligned}E[f(Y)] &\approx f(\mu) + f'(\mu)(E(Y) - \mu) = f(\mu) \\ \text{Var}[f(Y)] &\approx f'(\mu)^2 \text{Var}(Y) = f'(\mu)^2 \sigma^2\end{aligned}$$

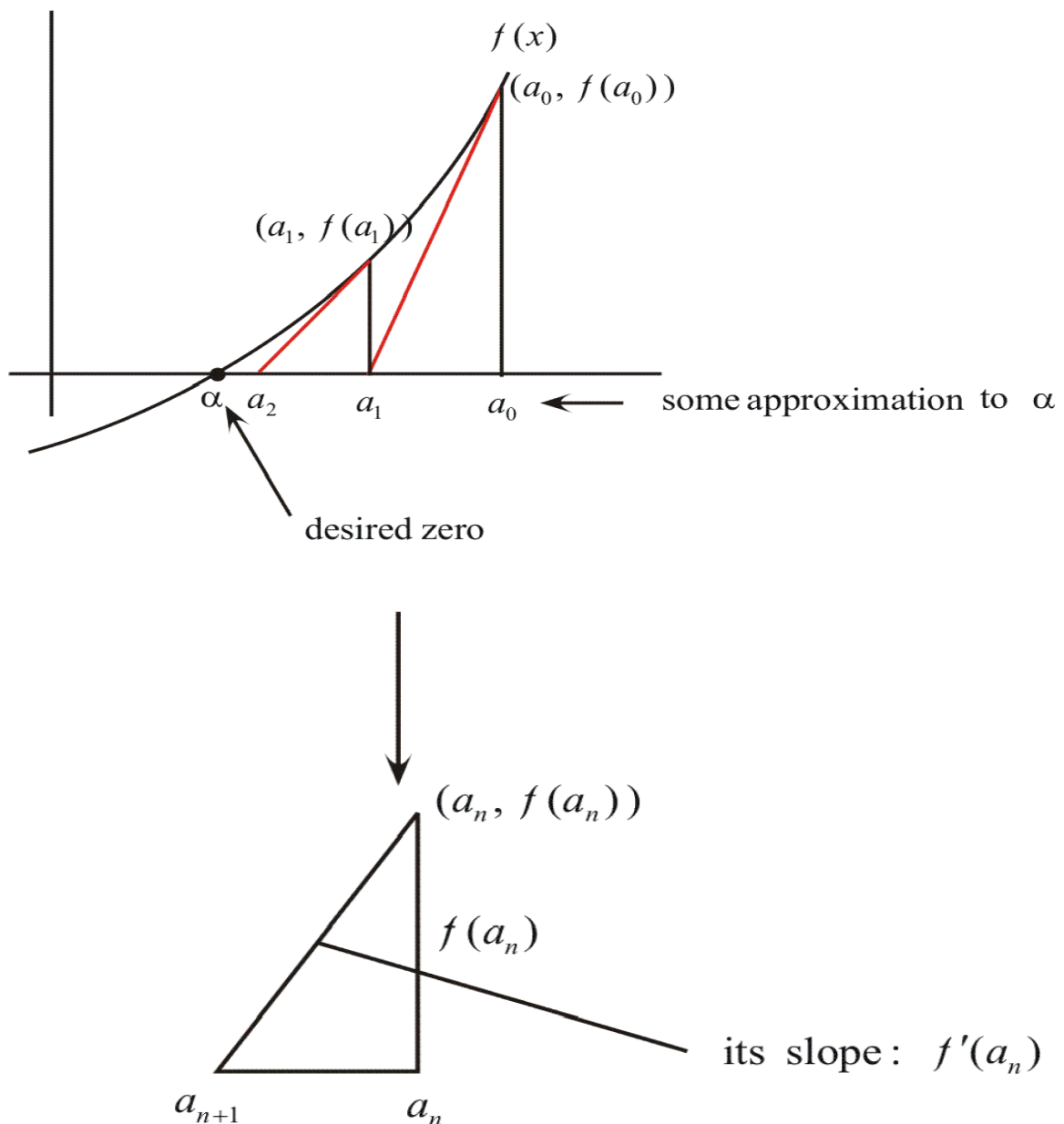
가 된다.

그렇다면 delta method는 왜 필요한 것일까? 만약 μ 가 아닌 $1/\mu$ 혹은 $\log(\mu)$ 를 추정하고자 할 때 이를 추정하기 위한 통계량으로 $1/\bar{X}$ 혹은 $\log(\bar{X})$ 를 사용한다면 매우 이상한 결과를 얻게될 것이다. X 자체에 대

한 분포만으로는 앞선 값들의 추정이 불가능하다. delta-method는 이러한 상황에서 $1/\bar{X}$ 과 $\log(\bar{X})$ 과 같은 통계량들의 분포를 유도하는 역할을 하기 때문에 중요한 개념이다.

(3) 뉴턴 방법

뉴턴-랩슨법(Newton-Raphson method)이라고도 불리는 **뉴턴 방법(Newton's method)**은 방정식의 해를 근사적으로 찾을 때 유용하게 사용되는 방법이다. 임의의 x 값에서 접선을 그은 후 접선이 x 축과 만나는 x 절편을 찾고, 그 점에서 또 다시 접선을 그어 x 축과 만나는 x 절편을 찾아가는 이러한 과정을 반복하여 점진적으로 해를 찾는 방법이다. 그림을 통해 더 자세히 이해해보자.



a_0 에서 출발하여 $f(x)$ 의 해인 α 를 근사적으로 찾는 과정을 살펴보았다. 위의 그림을 식으로 표현해보면 다음과 같이 일반화할 수 있다.

$$f'(a_n) = \frac{f(a_n)}{a_n - a_{n+1}}$$

$$a_{n+1} = a_n - \frac{f(a_n)}{f'(a_n)} \dots (*_1)$$

$$\lim_{n \rightarrow \infty} a_n \approx \alpha$$

• 오차해석

그렇다면 수열 a_n 은 얼마나 빠른 속도로 $f(x)$ 의 해에 수렴하게 될까? 함수 $f(x)$ 의 해를 α 라 하고, $f(\alpha)$ 를 a_n 근처에서 테일러 정리를 이용하여 2차항까지만 근사하면,

$$f(\alpha) = f(a_n) + (\alpha - a_n)f'(a_n) + \frac{1}{2}(\alpha - a_n)^2 f''(a_n)$$

가 되고, $f(\alpha)$ 에 0을 대입한 후 양변을 $f'(a_n)$ 으로 나누면,

$$0 = \frac{f(a_n)}{f'(a_n)} + (\alpha - a_n) + \frac{f''(a_n)}{2f'(a_n)}(\alpha - a_n)^2$$

위의 식에서 $(*_1)$ 식을 더하여 정리하면,

$$a_{n+1} = a_n + (\alpha - a_n) + \frac{f''(a_n)}{2f'(a_n)}(\alpha - a_n)^2$$

$$\alpha - a_{n+1} = \frac{-f''(a_n)}{2f'(a_n)}(\alpha - a_n)^2$$

오차를 $\epsilon_n = \alpha - a_n$ 이라 하고, 위의 식을 다시 정리하면 뉴턴 방법에서의 오차를 최종적으로 표현할 수 있다.

$$\epsilon_{n+1} = \left[\frac{-f''(a_n)}{2f'(a_n)} \right] \epsilon_n^2 \dots (*_2)$$

따라서 뉴턴 방법을 시행할수록 오차가 제곱에 비례해서 줄어드는 것을 알 수 있다. 그러나 시작점 a_0 를 잘못 정하면 반대로 오차가 제곱에 비례하여 커지는 문제가 생길 수도 있다. 이번에는 뉴턴 방법이 해에 제대로 수렴할 조건을 살펴보자.

$$\frac{-f''(a_n)}{2f'(a_n)} \approx \frac{-f''(\alpha)}{2f'(\alpha)} \equiv M \dots (*_3)$$

$(*_2)$ 식의 양변에 M 을 곱하여 정리하면 언제 오차가 감소하는지 알 수 있다.

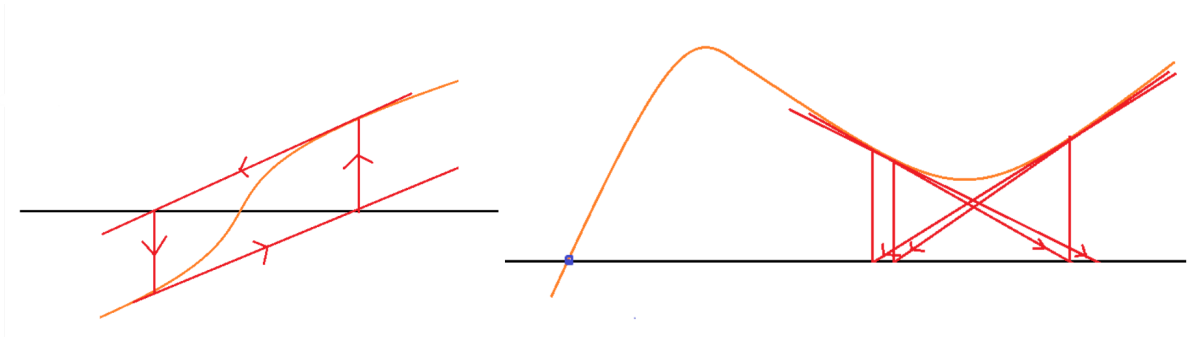
$$M\epsilon_{n+1} \approx [M\epsilon_n]^2 \dots (*_4)$$

모든 반복항이 $(*_3)$ 과 같이 α 에 가깝다면, $(*_4)$ 식을 점화식으로 사용하여 $\alpha - a_n$ 이 수렴하기 위한 조건을 알 수 있다.

$$\begin{aligned}
M\epsilon_n &\approx [M\epsilon_{n-1}]^2 \approx [M\epsilon_{n-2}]^{2^2} \approx [M\epsilon_{n-3}]^{2^3} \\
M\epsilon_n &\approx [M\epsilon_0]^{2^n} \\
|M(\alpha - a_0)| &< 1 \\
|\alpha - a_0| &< \frac{1}{|M|} = \left| \frac{2f'(\alpha)}{f''(\alpha)} \right|
\end{aligned}$$

• 뉴턴 방법의 장단점

뉴턴 방법은 오차가 제곱에 비례해서 줄어들기 때문에 몇 번만 시행해도 매우 빠른 속도로 수렴한다는 장점이 있다. 그러나 단점은 미분값을 알고 있어야 한다는 점이다. 코딩을 한다면 루프를 조금만 돌려도 되는 대신, 미분값을 구하느라 하나의 루프 내에서 계산하는 시간이 오래 걸린다는 단점을 가지고 있다고 볼 수 있다. 또한 방정식의 해가 여러 개가 있을 경우 초기값에 따라 하나의 근으로만 수렴하게 되는 경우도 있을 수 있기 때문에 주의가 필요하다. 뉴턴 방법을 사용할 수 없는 경우를 살펴보면 다음과 같다.



왼쪽의 경우처럼 변곡점 부근에서 해를 갖는 경우, 뉴턴 방법은 무한 루프에 빠져 소용이 없어진다. 오른쪽의 경우처럼 초기값을 해의 가까운 곳에 설정하지 않게 되면, 그림처럼 다른 부근에서 무한히 왼쪽과 오른쪽을 번갈아가며 계산되는 현상이 발생한다.

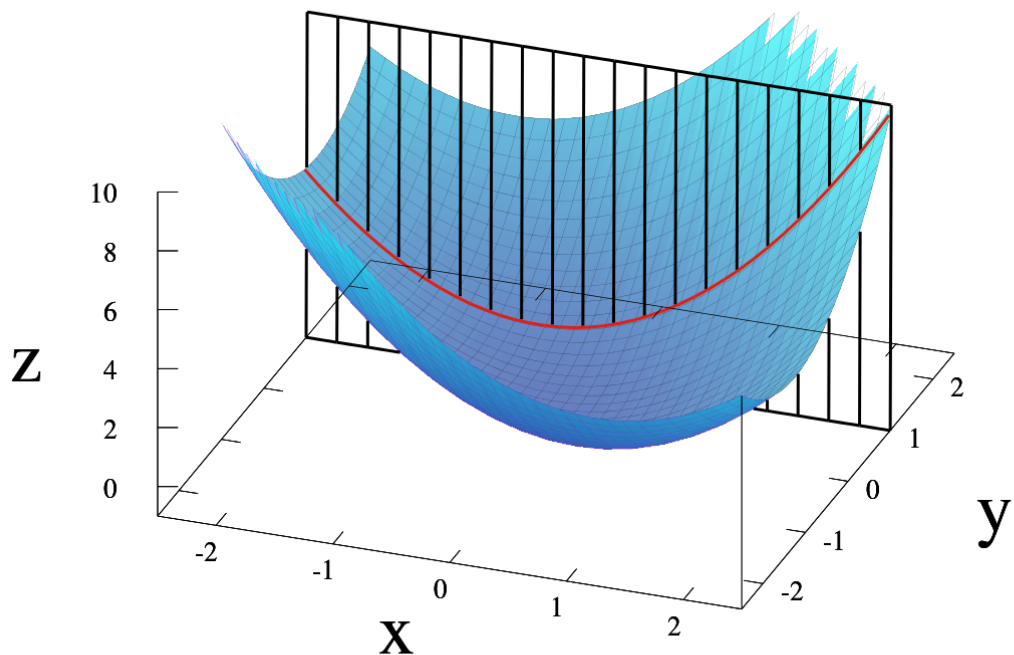
1.2. 편미분과 전미분

(1) 편미분

편미분은 다변수 함수에서 관심이 있는 한 변수만 변수로 생각하고, 나머지 변수들은 상수로 취급한 뒤 미분하는 방법을 말한다. 어떤 함수를 미분한 함수에 도함수라는 용어를 사용하듯, 어떤 함수를 편미분한 함수를 편도함수라고 한다.

$$f'(x, y, z, \dots) = \frac{\partial f}{\partial x} = \frac{\partial}{\partial x} f(x, y, z, \dots)$$

일변수 함수에서의 미분이 순간변화율이라는 의미를 갖고 있듯 다변수 함수에서의 편미분 역시 순간변화율의 의미를 갖고 있는 건 변함이 없으나, 한 가지 차이가 있다. 일변수 함수는 그래프 위의 점 하나에 대응하는 1개의 접선이 존재하지만, 다변수 함수는 무수히 많은 접선이 존재한다는 것이다. 기하학적으로 보면, 수많은 접선 중 하나의 접선을 구하기 위해 나머지 다른 변수의 값을 상수로 취급하여, 즉 관심있는 하나의 변수를 제외하고 나머지 변수들의 값을 고정한 뒤 미분을 하는 것이 편미분이다.



쉬운 예로, 국어, 영어, 수학 3개의 시험 과목의 평균을 구하는 함수 f 를 생각해보자. 이 때 국어 점수만 변화했을 때 전체 평균이 얼마나 변하는지 살펴보고자 한다면 f 를 국어에 대해 편미분하면 된다. 국어 과목이 1점 떨어질 때마다 평균이 0.33점씩 떨어짐을 알 수 있다.

$$f(\text{국어}, \text{영어}, \text{수학}) = \frac{\text{국어} + \text{영어} + \text{수학}}{3}$$

$$f'_x(\text{국어}, \text{영어}, \text{수학}) = \frac{1}{3}$$

그렇다면 국어, 영어, 수학의 점수가 모두 변화할 때 전체 평균이 얼마나 변하는지 알고 싶다면, 어떻게 하는 것이 좋을까? 이 때 사용하는 것이 바로 전미분이다.

(2) 전미분

전미분은 각 변수에 대한 편미분의 합으로 정의된다. 식으로 나타내보면 다음과 같다.

$$df(x, y, z, \dots) = \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy + \frac{\partial f}{\partial z}dz$$

이 때 dx, dy, dz, \dots 는 증분이라는 변화량을 나타내며, 어떤 함수를 전미분한 함수를 전도함수라고 한다. 모든 변수에 대해 미분하는 것이기 때문에 특별한 변수 표기 없이 함수 이름 앞에 d를 붙여 전도함수를 표현한다.

앞에서 살펴본 국어, 영어, 수학의 평균을 구하는 예제로 다시 돌아가보자. 만약 국어 점수가 6점, 영어 점수가 12점, 수학 점수가 30점 떨어졌다면 평균은 16점 떨어지게 될 것이다. 다음의 전미분한 식을 통해 구하면 된다.

$$df(\text{국어}, \text{영어}, \text{수학}) = \frac{1}{3}d\text{국어} + \frac{1}{3}d\text{영어} + \frac{1}{3}d\text{수학}$$

• 편미분에서의 chain rule

연쇄법칙(chain rule)이란 합성함수를 미분할 때 사용된다는 점을 기억할 것이다. 고등학교 때 다루는 일변수 함수의 연쇄법칙을 다시 한 번 살펴보도록 하자. 두 함수 $y = f(x)$ 와 $x = g(t)$ 가 주어지고, 두 함수 모두 미분가능할 때, y 를 t 에 대해 미분하면 다음과 같다.

$$\frac{dy}{dt} = \frac{dy}{dx} \frac{dx}{dt}$$

이 개념을 확장하여 다변수 함수의 여러 형태에도 적용할 수 있다. 확장시킨 2가지 형태를 살펴보고, 일반화된 연쇄법칙이 어떤 형태인지 알아보기로 한다.

Case1)

$z = f(x, y)$ 가 x, y 에 대해 미분가능하고 $x = g(t)$ 이고 $y = h(t)$ 두 함수 모두 t 에 대해 미분가능하다고 하자. 그러면 함수 $z = f(x, y)$ 는 변수 t 에 대해 다음과 같이 미분가능하다.

$$\frac{dz}{dt} = \frac{\partial f}{\partial x} \frac{dx}{dt} + \frac{\partial f}{\partial y} \frac{dy}{dt}$$

Case2)

함수 $z = f(x, y)$ 가 x, y 에 대해 미분가능하고 $x = g(s, t)$ 이고 $y = h(s, t)$ 두 함수 모두 s, t 에 대해 미분가능하다고 하자. 그러면 함수 $z = f(x, y)$ 는 변수 s, t 에 대해 다음과 같이 미분가능하다.

$$\begin{aligned} \frac{\partial z}{\partial s} &= \frac{\partial f}{\partial x} \frac{\partial x}{\partial s} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial s} \\ \frac{\partial z}{\partial t} &= \frac{\partial f}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial t} \end{aligned}$$

Case3) General chain rule

함수 $z = f(x_1, x_2, \dots, x_n)$ 에 n 개의 변수 (x_1, x_2, \dots, x_n) 에 대해 미분가능하고, 각 변수 $x_i = g_i(t_1, t_2, \dots, t_m)$ 가 m 개의 변수 (t_1, t_2, \dots, t_m) 에 대해 미분가능하다고 하자. 그러면 함수 $z = f(x_1, x_2, \dots, x_n)$ 은 m 개의 변수 (t_1, t_2, \dots, t_m) 에 대해 미분가능하다. 이 때 각 $i = 1, 2, \dots, m$ 에 대해 다음의 식이 성립한다.

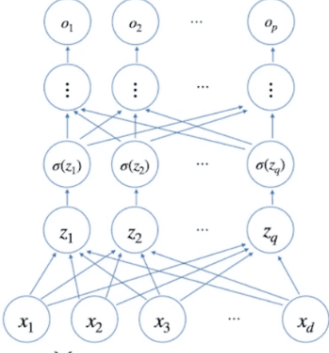
$$\frac{\partial z}{\partial t_i} = \frac{\partial f}{\partial x_1} \frac{\partial x_1}{\partial t_i} + \frac{\partial f}{\partial x_2} \frac{\partial x_2}{\partial t_i} + \dots + \frac{\partial f}{\partial x_n} \frac{\partial x_n}{\partial t_i}$$

• 딥러닝에서의 역전파(backward propagation) 알고리즘

딥러닝의 작동 원리는 크게 순전파 알고리즘과 역전파 알고리즘으로 구분된다. 순전파(forward propagation)는 입력값 X 를 받아 선형모델과 활성화수를 반복적으로 적용하여 출력하는 연산이다. 이 때, 가중치에 해당하는 W 를 학습시키기 위해서는 각 가중치에 대한 gradient 벡터를 계산이 필요한데, 역전파 알고리즘이 이 과정을 수행한다. 손실함수(loss function)를 이용하여 각 층에 존재하는 W 들에 대한 미분값을 계산한 뒤 이들을 가지고 W 값을 업데이트 시키게 되는데, 연쇄 법칙을 이용하여 연산을 수행한다. 따라서 역전파 알고리즘은 역순차적으로 층마다 미분값을 계산하여 적용하는 과정이라고 볼 수 있다. 이와 관련해서는 수학2팀에서 자세히 다뤄질 것이기 때문에 여기서는 가볍게 살펴보고 지나가기로 한다.

각 층에서 계산된 gradient 벡터들은 밑의 층으로 전달되는 형태이며, 아래층에 있는 gradient 벡터를 계산하기 위해서는 위층에 있는 gradient 벡터가 필요하기 때문에 위층에서 아래층으로 내려오면서 업데이트하는 방식을 취한다. 연쇄법칙을 이용하여 가중치 W 가 업데이트되는 과정을 나타내보면 다음과 같다.

<순전파>



MLP의 패러미터는 L 개의 가중치 행렬 $\mathbf{W}^{(L)}, \dots, \mathbf{W}^{(1)}$ 과 로 이루어져 있다 $\mathbf{b}^{(L)}, \dots, \mathbf{b}^{(1)}$

$$\mathbf{H}^{(\ell)} = \sigma(\mathbf{Z}^{(\ell)})$$

$$\mathbf{Z}^{(\ell)} = \mathbf{H}^{(\ell-1)}\mathbf{W}^{(\ell)} + \mathbf{b}^{(\ell)}$$

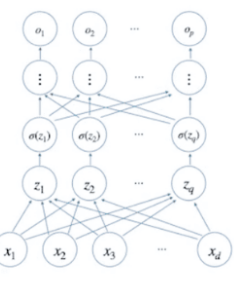
$$\mathbf{H}^{(1)} = \sigma(\mathbf{Z}^{(1)})$$

$$\mathbf{Z}^{(1)} = \mathbf{X}\mathbf{W}^{(1)} + \mathbf{b}^{(1)}$$

boostcamp AI Tech © NAVER Connect Foundation

13

<역전파>



손실함수를 \mathcal{L} 이라 했을 때 역전파는 $\partial\mathcal{L}/\partial\mathbf{W}^{(\ell)}$ 정보를 계산할 때 사용된다

$$\mathbf{O} = \mathbf{Z}^{(L)}$$

$$\mathbf{H}^{(\ell)} = \sigma(\mathbf{Z}^{(\ell)})$$

$$\mathbf{Z}^{(\ell)} = \mathbf{H}^{(\ell-1)}\mathbf{W}^{(\ell)} + \mathbf{b}^{(\ell)}$$

$$\mathbf{H}^{(1)} = \sigma(\mathbf{Z}^{(1)})$$

$$\mathbf{Z}^{(1)} = \mathbf{X}\mathbf{W}^{(1)} + \mathbf{b}^{(1)}$$

boostcamp AI Tech © NAVER Connect Foundation

17

< l 번째 층에서의 연쇄법칙을 활용한 역전파 알고리즘의 계산 >

$$\frac{\partial L}{\partial W^{(l)}} = \frac{\partial L}{\partial O} \times \dots \times \frac{\partial Z^{(l)}}{\partial W^{(l)}}$$

$$\frac{\partial L}{\partial b^{(l)}} = \frac{\partial L}{\partial O} \times \dots \times \frac{\partial Z^{(l)}}{\partial b^{(l)}}$$

- Multiple delta-method

\vec{Y} 를 $E(\vec{Y}) = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$ 와 $Var(\vec{Y}) = \Sigma$ 를 갖는 랜덤 벡터라 하고, $f(\vec{Y})$ 가 \vec{Y} 에 대해 미분 가능한 함수라 하자. 그러면,

$$Var(f(\vec{Y})) = \left(\frac{\partial f(\vec{u})}{\partial \vec{u}} \right)^T \Sigma \left(\frac{\partial f(\vec{u})}{\partial \vec{u}} \right)$$

$$= \left(\frac{\partial f(\vec{u})}{\partial \mu_1}, \frac{\partial f(\vec{u})}{\partial \mu_2} \right) \Sigma \begin{pmatrix} \frac{\partial f(\vec{u})}{\partial \mu_1} \\ \frac{\partial f(\vec{u})}{\partial \mu_2} \end{pmatrix}$$

- 상대위험률(Relative Risk)과 오즈비(Odds Ratio)

다음의 2차원 분할표(2×2 contingency table)를 통해 상대위험률과 오즈비의 개념을 정리해보자.

<상대위험률>

	1	2	
1	$n_{11}(\pi_1)$	$n_{12}(1 - \pi_1)$	$n_{1+}(1)$
2	$n_{21}(\pi_2)$	$n_{22}(1 - \pi_2)$	$n_{2+}(1)$
	n_{+1}	n_{+2}	n

$$RR = \frac{\pi_{11}}{\pi_{12}} = \frac{n_{11}/n_{1+}}{n_{21}/n_{2+}}$$

<오즈비>

	1	2	
1	$n_{11}(\pi_1)$	$n_{12}(\pi_2)$	n_{1+}
2	$n_{21}(1 - \pi_1)$	$n_{22}(1 - \pi_2)$	n_{2+}
	$n_{+1}(1)$	$n_{+2}(1)$	n

$$OR = \frac{n_{11}/n_{12}}{n_{21}/n_{22}} = \frac{n_{11}n_{22}}{n_{12}n_{21}}$$

상대위험률은 n_{1+} 와 n_{2+} 와 같이 행합계가 고정된 코호트 연구(cohort study)나 임상시험(clinical trial)에서 주로 사용된다. 예를 들어, 아스피린 복용이 심장병 발병 위험을 낮추는 데에 효과가 있는지 알아보기 위해 약 60개월 간 관찰하여 실험을 진행하는 경우이다. 일정 기간이 지난 후에 효과가 있는지 없는지를 알아보는 것이기 때문에 후향적 연구(prospective design)이다.

이와 달리, 오즈비는 n_{+1} 과 n_{+2} 와 같이 열합계가 고정된 사례-대조군 연구(case-control study)에서 주로 활용된다. 심장발작을 일으킨 환자와 그렇지 않은 사람들을 조사하여 과거에 약물남용을 했었는지 아닌지를 조사하는 경우가 그 예이다. 일정 기간이 지난 후의 효과를 확인하고자 하는 것이 아니라, 현재 상태를 역추적하여 과거의 경험에 대해 조사하는 것이기 때문에 전향적 연구(retrospective design)이다.

따라서 상황에 따라 상대위험률과 오즈비 중 어느 것을 사용할지 판단하는 것은 중요한 문제이다. 특히 바이오 분야에서 그러한데, 이와 관련하여 SAS와 함께 더 자세히 공부해보고 싶다면, 김동욱 교수님의 바이오통계입문 강의를 수강하기 바란다^^

이렇게까지 열심히 상대위험률과 오즈비에 대해 설명한 이유는 앞서 설명한 multivariate delta-method를 활용하면 분산을 쉽게 구해냄으로써 로그 상대위험률과 로그 오즈비에 대한 신뢰구간을 도출해낼 수 있기 때문이다. 각 경우에 대해 살펴보기로 한다.

<상대위험률>

$$\begin{aligned} f(\pi_1, \pi_2) &= \log \frac{\pi_1}{\pi_2} = \log \pi_1 - \log \pi_2 \\ \frac{\partial f(\pi_1, \pi_2)}{\partial \pi_1} &= \frac{1}{\pi_1} \quad \frac{\partial f(\pi_1, \pi_2)}{\partial \pi_2} = -\frac{1}{\pi_2} \\ \text{Var}(\log \frac{p_1}{p_2}) &= \left(\frac{1}{\pi_1}, -\frac{1}{\pi_2} \right) \begin{pmatrix} \frac{\pi_1(1-\pi_1)}{n_{1+}} & 0 \\ 0 & \frac{\pi_2(1-\pi_2)}{n_{2+}} \end{pmatrix} \begin{pmatrix} \frac{1}{\pi_1} \\ -\frac{1}{\pi_2} \end{pmatrix} = \frac{1-p_1}{n_{1+}p_1} + \frac{1-p_2}{n_{2+}p_2} \\ &= \frac{1 - n_{11}/n_{1+}}{n_{11}} + \frac{1 - n_{21}/n_{2+}}{n_{21}} \\ &= \frac{1}{n_{11}} - \frac{1}{n_{1+}} + \frac{1}{n_{21}} - \frac{1}{n_{2+}} \end{aligned}$$

이를 바탕으로 상대위험률에 대한 $100(1 - \alpha)$ 신뢰구간을 구하면,

$$\exp(\log RR \pm Z_{\alpha/2} \sqrt{\text{Var}(\log RR)}) = RR \times \exp(\pm Z_{\alpha/2} \sqrt{\text{Var}(\log RR)}) \text{가 된다.}$$

<오즈비>

$$\begin{aligned} OR &= \frac{n_{11}/n_{12}}{n_{21}/n_{22}} \\ &= (\log n_{11} - \log n_{12}) - (\log n_{21} - \log n_{22}) \\ &= \log \frac{n_{11}}{n_{12}} - \log \frac{n_{21}}{n_{22}} \\ \text{Var}(\log \frac{n_{11}}{n_{12}} - \log \frac{n_{21}}{n_{22}}) &= \frac{1}{n_{11}} + \frac{1}{n_{12}} + \frac{1}{n_{21}} + \frac{1}{n_{22}} \end{aligned}$$

이를 바탕으로 오즈비에 대한 $100(1 - \alpha)$ 신뢰구간을 구하면,

$$\exp(\log OR \pm Z_{\alpha/2} \sqrt{\text{Var}(\log OR)}) = OR \times \exp(\pm Z_{\alpha/2} \sqrt{\text{Var}(\log OR)}) \text{가 된다.}$$

Reference

** 선형근사와 테일러 전개, 뉴턴 방법

(1) 선형근사

James Stewart (2015) Calculus, 8th ed., Cengage, 188

10. 선형근사 (Linear Approximation)

과학이나 공학에서는 때때로 정확한 값 보다는 적은 노력으로 꽤 근접한 유사값을 찾아 낼 수 있다면 그것을 높이 평가하기도 한다. 쉬운 예로 $y = \sin\{x\}$ 가 $x=0$ 에서 $\sin\{0\} = 0$ 임은 알지만 $\sin\{0.2\}$.

☞ <https://vegatrash.tistory.com/19>

10 선형근사

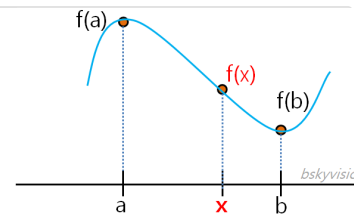
선형보간법(linear interpolation)과 삼차보간법(cubic interpolation), 제대로 이해하자

선형보간법(linear interpolation)에 대해서는 잘 설명된 자료가 많지만, 삼차보간법(cubic interpolation)에 대해서는 읽을 만한 관측은 자료를 찾기가 쉽지 않습니다. 아마도 삼차보간법에 대해 글을 쓰신 분들도 완벽하게 원리를 이해하고 쓴 것이 아닌 것 같습니다. 아인슈타인이 이렇게

☞ <https://bskyvision.com/entry/%EC%84%A0%ED%98%95%EB%B3%B4%EA%B0%84%EB%B2%95linear-interpolation%EA%B3%BC-%EC%82%BC%EC%B0%A8%EB%B3%B4%EA%B0%84%EB%B2%95cubic-interpolation-%EC%9D%B4%EB%B3%B4%EB%88%9C-%EC%97%86%EB%8B%A4>

(2) 테일러 전개

James Stewart (2015) Calculus, 8th ed., Cengage, 799



[손으로 푸는 통계] #11. 적률생성함수, MGF (중심극한정리를 위한 재료 #2)

글이 더 편하신 분 <http://hsm-edu.tistory.com/23>

☞ <https://www.youtube.com/watch?v=gubleOnA5ys&t=246s>

신개념 통계 문맹초!!

t-test ANOVA wilcoxon signed rank test

손으로만 푸는 통계

11. 적률생성함수, MGF

(중심극한정리 증명을 위한 재료 #2)

이근백교수님 (2019) 범주형자료분석 교안 중 delta-method

https://bayestour.github.io/blog/2019/07/02/sobtest_2.html

(3) 뉴턴 방법

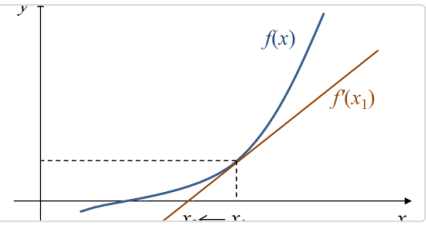
James Stewart (2015) Calculus, 8th ed., Cengage, 272

좌준수교수님(2019) 해석학1 교안 중 Newton's method

뉴턴법/뉴턴-랩슨법의 이해와 활용(Newton's method)

뉴턴법/뉴턴-랩슨법 하면 대부분 방정식의 근사해를 구하는 방법 정도로 알고 있지만 뉴턴법을 확장하면 연립방정식의 해, 나아가서는 비선형(non-linear) 모델의 파라미터를 구하는 문제까지 확장될 수 있습니다. 뉴턴법/뉴턴랩슨법 뿐만 아니라 가우스-뉴턴법,

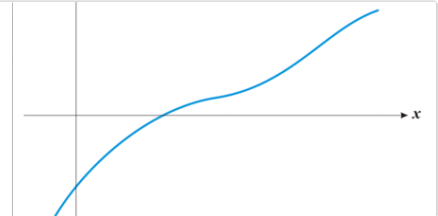
🔗 <https://darkpgmr.tistory.com/58>



뉴턴의 방법과 오차해석 (Newton's method)

방정식을 풀지 않고, 근이 대략 어느 정도 숫자인지 추정하는 방법이 있습니다. 그중에서 뉴턴의 방법(Newton's method)을 소개하려고 합니다. 뉴턴 방법은 어떤 점에서 함수값과 기울기를 알 때, 그 함수를 일차..

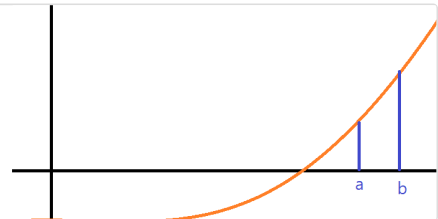
🔗 <https://phy64ev1.tistory.com/20>



수치해석 - Newton-Raphson 방법

손담 • 2020. 8. 22. 18:34 Newton-Raphson은 개방법의 일종이다. 자, 방정식의 근을 구하는 방법은 크게 두가지가 있었다. 먼저 근을 포함하는 구간 내에서 2개의 초기값에 기초하여 근을 구하는 방법이 구간법이다. 오늘 알아볼 Newton-Raphson방법은 개

🔗 <https://blog.naver.com/PostView.nhn?blogId=ptm0228&logNo=222067871109>



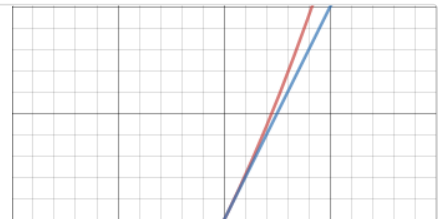
** 편미분과 전미분

James Stewart (2015) Calculus, 8th ed., Cengage, 953-955

[딥러닝 with 수학] 3편 - 편미분, 전미분

$f(x, y)$ 를 x 에 대해 편미분하면 x 만 변수로 취급되고, y 는 상수로 취급됩니다. 따라서 y , 1 이 두 항은 상수항으로 처리되어 미분하면 사라지고, xy , x 두 항을 미분하면 각각 y , 1 이 되어 편도함수가 $y+1$ 이 됩니다. y 에 대해 편미분하면 반대로 y 만 변수로 취급되고 x

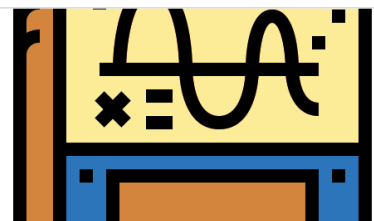
🔗 <https://m.blog.naver.com/kmc7468/221898862253>



미적분학 - 다변수 함수의 연쇄법칙

안녕하세요. 지난 포스팅의 미적분학 - 다변수 함수의 미분가능성에서는 다변수 함수의 편미분이 존재한다고 해서 미분이 가능하지 않다는 점과 미분가능성에 대한 명확한 정의 그리고 전미분(total derivative)에..

🔗 <https://everyday-image-processing.tistory.com/348>



딥러닝의 기초적 이해 - 선형모델부터 역전파에 이르기까지

딥러닝의 학습방법 by 임성빈 교수님, BoostCamp AI Tech 2주차

🔗 https://blogik.netlify.app/BoostCamp/U_stage/07_deep_learning_basic/

$$\nabla_{W^{(1)}} \mathcal{L} = (\nabla_{W^{(1)}} \mathcal{L})(\nabla_{\mathbf{h}} \mathbf{h})(\nabla_{\mathbf{h}} \mathbf{h})(\nabla_{\mathbf{z}} \mathbf{z})(\nabla_{\mathbf{z}} \mathbf{z})(\nabla_{\mathbf{z}} \mathbf{z}) \longleftrightarrow \frac{\partial \mathcal{L}}{\partial W_{ij}^{(1)}} = \sum_{r,k} \frac{\partial \mathcal{L}}{\partial o_i} \frac{\partial o_i}{\partial h_r} \frac{\partial h_r}{\partial z_k} \frac{\partial z_k}{\partial W_{ij}^{(1)}}$$

$$\mathbf{o} = \mathbf{W}^{(2)} \mathbf{h} + \mathbf{b}^{(2)}$$

$$\mathbf{h} = \sigma(\mathbf{z})$$

$$\mathbf{z} = \mathbf{W}^{(1)} \mathbf{x} + \mathbf{b}^{(1)}$$

$$\frac{\partial h_r}{\partial z_k} = \sigma'(z_k) \delta_{rk}$$

$$\frac{\partial z_k}{\partial W_{ij}^{(1)}} = \frac{\partial}{\partial W_{ij}^{(1)}} \sum_r x_r W_{rk}^{(1)} = x_i \delta_{jk}$$

김동욱교수님 (2022) 바이오통계입문 교안 중 relative risk and log odds ratio

이근백교수님 (2019) 범주형자료분석 교안 중 multivariate delta-method, standard error of log relative risk and log odds ratio

