# 4-1. 적분의 정의 : 미적분의 기본 정리(Fundamental Theorem of Calculus)

적분과 미적분의 기본 정리는 수학에서 매우 중요한 개념입니다. 이 정리는 미분과 적분이 밀접하게 연결되어 있음을 보여주며, 미적분학의 핵심적인 이론 중 하나로 꼽힙니다.

1. 적분의 정의

   적분은 주어진 함수의 구간 내에서 함수 값의 누적된 합을 구하는 과정입니다. 이를 통해 우리는 곡선 아래의 면적을 계산할 수 있습니다. 적분에는 두 가지 중요한 종류가 있습니다.

   - 정적분(Definite Intergral): 함수 $f(x)$에 대해, 두 점 a와 b 사이의 구간에서 곡선 아래의 면적을 구하는 것이 정적분입니다. 기호로는 다음과같이 나타냅니다

   $$\int_a^b f(x)dx$$

   여기서 $dx$는 적분 변수 $x$에 대한 미소한 변화량을 의미하면, $f(x)$는 적분할 함수입니다.

   - 부정적분(Indefinite Integral): 부정적분은 정적분과 달리 구간이 정해져 있지 않으며, 미분의 역연산입니다. 함수$f(x)$의 부정적분은 미분했을 ㄸ 원래 함수로 돌아오는 함수 $F(x)$입니다. 기 호로는 다음과 같이 나타냅니다.

   $$\int f(x)dx = F(x) + C$$

   여기서 $F(x)$는 $f(x)$의 원시함수(antiderivative)이며, $C$는 상수입니다.

2. 미적분의 기본 정리(Fundamental Theorem of Calculus)

   미적분의 기본 정리는 미분과 적분 사이의 관계를 설명하는 두 부분으로 구성됩니다.

   첫 번째 기본 정리

첫 번째 기본 정리는 정적분과 미분이 서로 역연산임을 보여줍니다. 구체적으로, 함수$f(x)$가구간 $[a, b]$에서 연속이고, $F(x)$가 $f(x)$의 원시함수라면 다음과 같은 관계가 성립합니다.

$$\int_a^b f(x)dx = F(b) - F(a)$$

이 정리는 구간 $[a, b]$에서 적분을 계산할 때, 함수 $f(x)$의 원시함수를 구하고 $b$에서의 값과 $a$에서의 값을 빼면 된다는 것을 의미합니다. 즉, 곡선 아래의 면적을 구하는 문제를 원시함수를 이용해서 해결할 수 있게 해줍니다.

두 번째 기본 정리

두 번째 기본 정리는 적분의 결과가 다시 미분을 통해 원래 함수로 돌아갈 수 있음을 보여줍니다. 만약 함수 $f(x)$가 구간$[a, b]$에서 연속이고, 함수 $g(x)$가 다음과 같이 정의된다면

$$g(x) = \int_a^x f(t)dt$$

이때 함수 $g(x)$를 미분하면 원래의 함수 $f(x)$가 나옵니다.

$$g'(x) = f(x)$$

이 정리는 적분과 미분이 긴밀히 연결되어 있음을 보여줍니다. 적분을 통해 함수의 누적 값을 구한 후, 그 결과를 미분하면 다시 원래 함수로 되돌아간다는 의미입니다.

3. 적분과 딥러닝

딥러닝에서 적분 자체가 자주 직접적으로 쓰이지는 않지만, 연속적인 함수에 대한 직관과 미분에 대한 이해를 강화하는데 도움이 됩니다. 미적분의 기본 정리는 딥러닝에서 주로 사용하는 미분 개념과 연력되어 있어, 특히 역전파(Backpropagation)와 같은 알고리즘을 이해할 때 미적분의 이해가 중요한 역할을 합니다.

따라서 적분을 깊이 이해하는 것은, 딥러닝에서 미분과 최적화 문제를 다룰 때 유용할 수 있습니다.

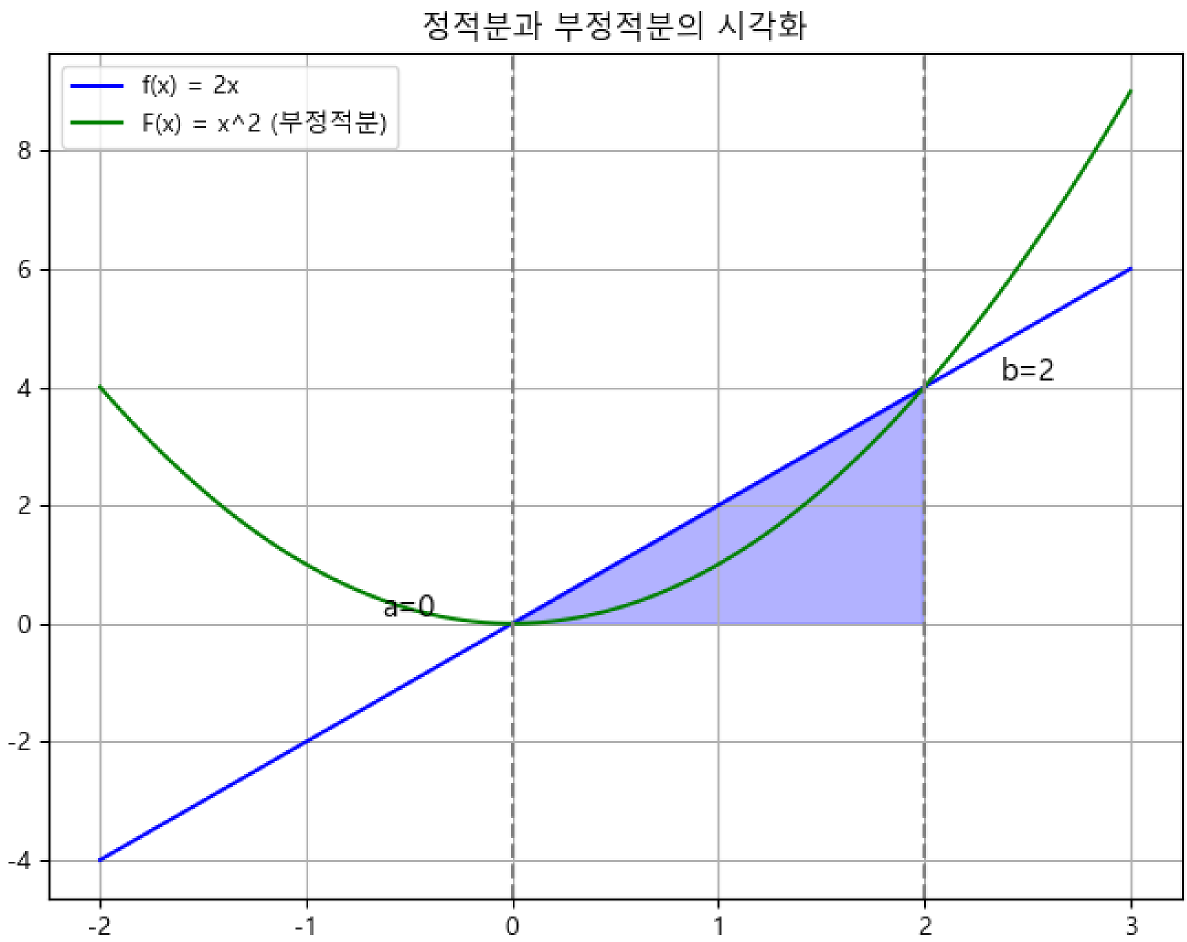
적분은 연속적인 변화에 대한 이해를 돕고, 이를 통해 더 나은 직관을 제공할 수 있습니다.

4. 파이썬 코드 및 그래프 시각화

- 2D 그래프 (정적분과 부정적분)

```
# 선형 독립성(Linear Independence)
# 한글 폰트 설정을 추가하여 문제 해결
import matplotlib.pyplot as plt
import matplotlib.font_manager as fm
import numpy as np
from scipy.integrate import quad


# 한글 폰트 설정
plt.rcParams['font.family'] = 'AppleGothic' if 'AppleGothic'
plt.rcParams['axes.unicode_minus'] = False  # 마이너스 부호 깨짐

# 함수 정의
def f(x):
    return 2 * x

# 부정적분 함수 정의
def F(x):
    return x**2

# 정적분 계산
a = 0  # 하한
b = 2  # 상한
integral_value, _ = quad(f, a, b)

# 그래프 그리기
x = np.linspace(-2, 3, 100)
y = f(x)
y_integral = F(x)

plt.figure(figsize=(8, 6))

# 원래 함수 그래프 (2x)
```

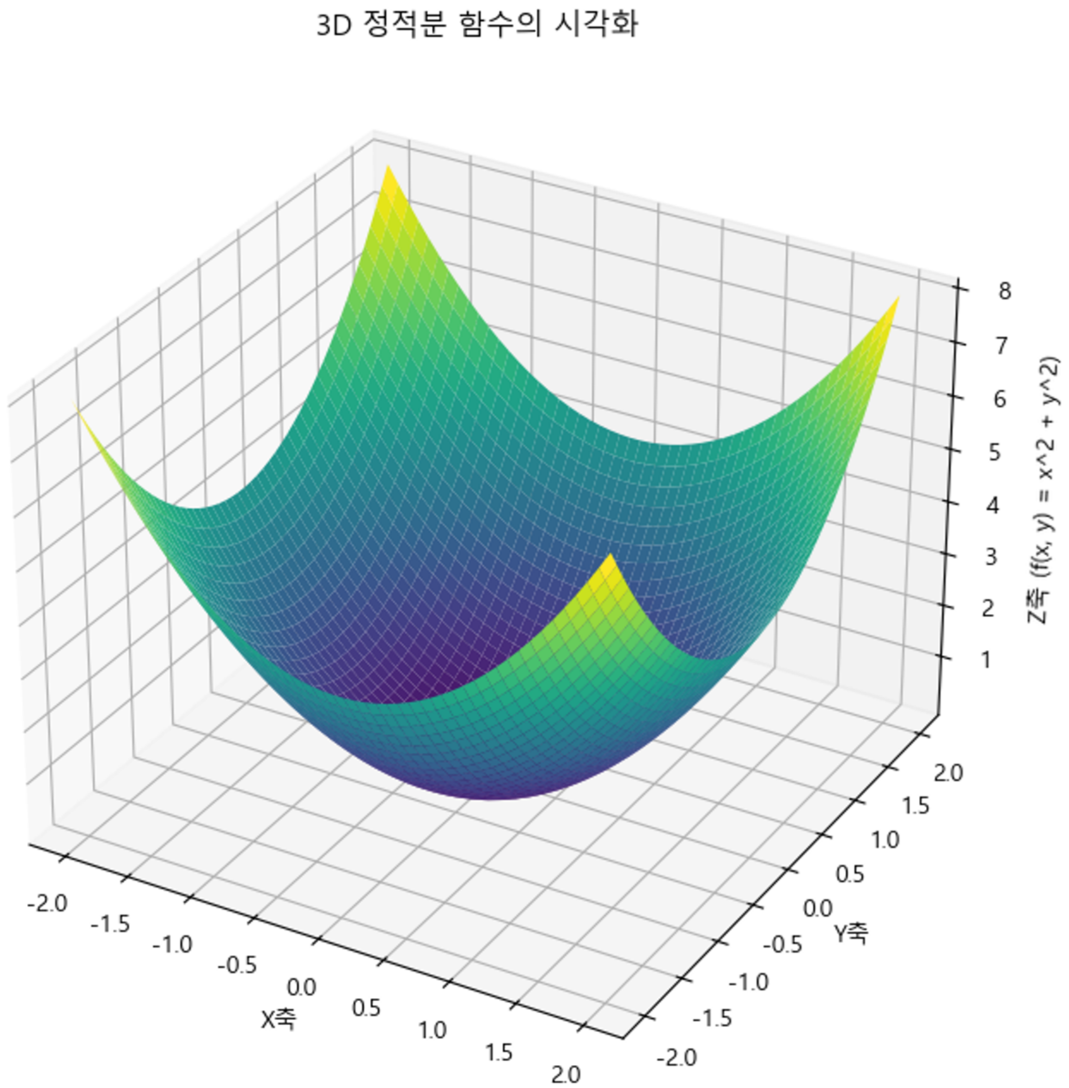
```
plt.plot(x, y, label='f(x) = 2x', color='blue')
plt.fill_between(x, y, where=[(a <= xi <= b) for xi in x], co

# 부정적분 함수 그래프 (x^2)
plt.plot(x, y_integral, label='F(x) = x^2 (부정적분)', color='g

# 그래프 라벨링
plt.title('정적분과 부정적분의 시각화')
plt.axvline(x=a, color='gray', linestyle='--')
plt.axvline(x=b, color='gray', linestyle='--')
plt.text(a-0.5, 0, f'a={a}', ha='center', va='bottom', fontsi
plt.text(b+0.5, 4, f'b={b}', ha='center', va='bottom', fontsi
plt.legend()
plt.grid(True)
plt.show()

print(f"정적분 결과: {integral_value}")
```




- 3D 그래프 (정적분과 부정적분)

```
# 선형 독립성(Linear Independence)
# 한글 폰트 설정을 추가하여 문제 해결
import matplotlib.pyplot as plt
import matplotlib.font_manager as fm
import numpy as np
from scipy.integrate import quad


# 한글 폰트 설정
plt.rcParams['font.family'] = 'AppleGothic' if 'AppleGothic'
plt.rcParams['axes.unicode_minus'] = False  # 마이너스 부호 깨짐

# 3D 정적분 함수 정의
def f3d(x, y):
    return x**2 + y**2
```

```python
# 그래프 그리기
x = np.linspace(-2, 2, 50)
y = np.linspace(-2, 2, 50)
X, Y = np.meshgrid(x, y)
Z = f3d(X, Y)

fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# 3D 그래프 그리기
ax.plot_surface(X, Y, Z, cmap='viridis', edgecolor='none')

# 그래프 라벨링
ax.set_title('3D 정적분 함수의 시각화')
ax.set_xlabel('X축')
ax.set_ylabel('Y축')
ax.set_zlabel('Z축 (f(x, y) = x^2 + y^2)')

plt.show()
```




- 2D 그래프
  - 함수 $f(x) = 2x$의 그래프와 부정적분 $F(x) = x^2$의 그래프를 그립니다.
  - 정적분을 계산한 구간 [0, 2]의 곡선 아래 면적을 파란색 음영으로 표시했습니다.
- 3D 그래프
  - 함수 $f(x, y) = x^2 + y^2$의 3D 곡면을 시각화하여, 다차원에서의 적분된 값을 시각적으로 표현합니다.
  - plot_surface를 사용해서 3D 곡면을 생성하며, x축 , y축, z축을 라벨링합니다.