svm2

1. 문제 요약

5절의 SVM_02 실험과의 비교에서 차이가 발생하는 원인 규명해보자.

의문사항: SVM_03 예제는99.2% 정확도인데, SVM_02 예제에서는99.0% 정확도이다.
의심점: shuffle 과정에서 서로 다른 데이터를 학습데이터와 테스트데이터로 쓸 수도 있다.
반론: RandomState(1234)의seed 값이같다. 이는 KNN에서도 계속 써오던 값이다.

* 그동안의 추정: seed가 같으면 난수 발생 패턴도 같다. True or False?

Seed가 난수 발생의 재연을 하는가 하지 않는가?

2. 결론

seed가 같으면 난수 발생 패턴도 같다. 이 때문에 결과 값도 동일한 정확도인 99.0%로 측정된다.

3. `seed` 가 같을 때, 난수 발생 패턴을 확인하는 실험

1) 실험 개요

난수 생성기의 재현성을 확인하는 것이다. 이를 통해 동일한 시드(seed) 값을 사용했을 때 난수 생성 패턴이 동일하게 유지되는지, 그리고 다른 시드 값을 사용했을 때 난수 생성 패턴이 달라지는지 확인하고자 한다.

2)실험 방법

두 가지 경우에 대해 실험 수행:

ㄱ. 같은 시드 값을 사용하여 난수를 생성하고, 두 번의 실행 결과를 비교합니다.

ㄴ. 다른 시드 값을 사용하여 난수를 생성하고, 두 번의 실행 결과를 비교합니다.

실험에 사용된 코드는 다음과 같다.

- **같은 시드 값을 사용하는 경우**

```
import numpy as np

# 같은 시드를 사용하여 난수를 생성
seed = 1234
random_state1 = np.random.RandomState(seed)
random_numbers1 = random_state1.rand(5)

random_state2 = np.random.RandomState(seed)
random_numbers2 = random_state2.rand(5)
```

```
print("같은 시드를 사용한 첫 번째 난수 배열:", random_numbers1)
print("같은 시드를 사용한 두 번째 난수 배열:", random_numbers2)

# 결과 비교
same_seed_same_output = np.array_equal(random_numbers1, random_numbers
2)
print("같은 시드를 사용했을 때 결과가 동일한가요?", same_seed_same_output)
```

- **다른 시드 값을 사용하는 경우:**

```
# 다른 시드를 사용하여 난수를 생성
random_state3 = np.random.RandomState(5678)
random_numbers3 = random_state3.rand(5)

random_state4 = np.random.RandomState(8765)
random_numbers4 = random_state4.rand(5)

print("다른 시드를 사용한 첫 번째 난수 배열:", random_numbers3)
print("다른 시드를 사용한 두 번째 난수 배열:", random_numbers4)

# 결과 비교
different_seed_different_output = np.array_equal(random_numbers3, rand
om_numbers4)
print("다른 시드를 사용했을 때 결과가 다른가요?", different_seed_different_ou
tput)
```

3) 실험 결과

- **같은 시드를 사용한 경우**:

```
같은 시드를 사용한 첫 번째 난수 배열: [0.19151945 0.62210877 0.43772774 0.78535858 0.77997581]
같은 시드를 사용한 두 번째 난수 배열: [0.19151945 0.62210877 0.43772774 0.78535858 0.77997581]
같은 시드를 사용했을 때 결과가 동일한가요? True
```

- **다른 시드를 사용한 경우**:

```
다른 시드를 사용한 첫 번째 난수 배열: [0.48932698 0.05933244 0.36620243 0.51886544 0.59822501]
다른 시드를 사용한 두 번째 난수 배열: [0.03278543 0.80595138 0.66984421 0.76030572 0.27496366]
다른 시드를 사용했을 때 결과가 동일한가요? False
```

4) 결론

실험 결과, 동일한 시드 값을 사용하여 난수를 생성했을 때 항상 동일한 난수 배열이 생성됨을 확인할 수 있었다. 이는 난수 생성기의 재현성이 보장됨을 의미한다. 반면, 다른 시드 값을 사용했을 때 생성된 난수 배열은 서로 달랐다. 따라서, 시드 값이 같으면 난수 생성 패턴도 동일하다.

5) 참고 문헌

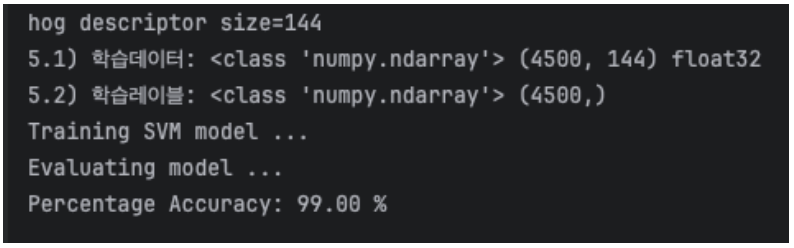
- NumPy 공식 문서:
  https://numpy.org/doc/stable/reference/random/legacy.html#numpy.random.RandomState

4. SVM_02와 SVM_03의 정확도

1) svm_02

아래와 같이 C=12.5, gamma=0.50625일 때, 정확도 99.00%가 출력된다.

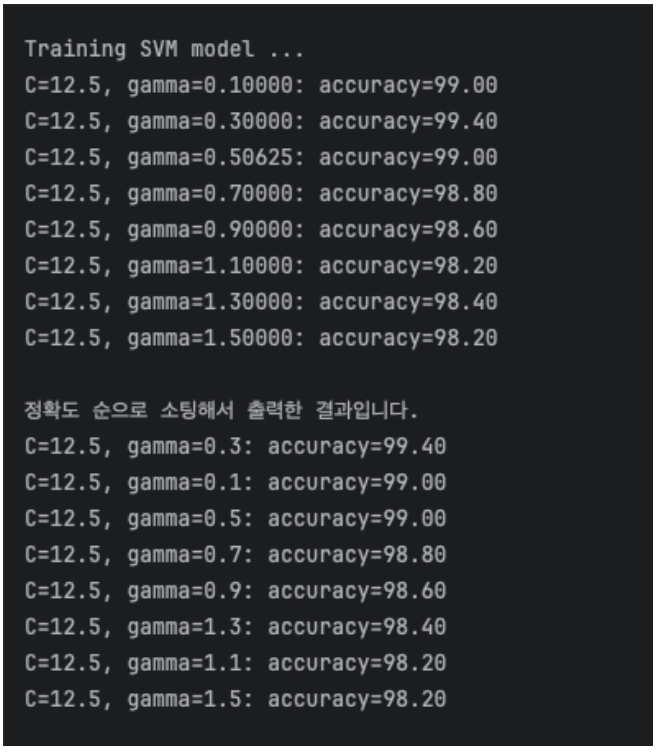
```
# 6) SVM 모델 객체를 하나 생성한다.
    print('Training SVM model ...')
    model = svm_init(C=12.5, gamma=0.50625)
```



2) svm_03

아래의 코드와 같이 C=12.5에 다양한 gamma 의 경우별로 나누어 정확도를 측정했다. 이 때의 결과 중 SVM_02 예제와의 비교를 위해 같은 조건 즉, C=12.5, gamma=0.50625에 해당하는 accuracy를 확인해보면 99.00%로 동일한 정확도를 출력한다. 더하여 C=12.5, gamma=0.3 일 때, 가장 큰 정확도인 99.40%를 출력하는 것을 확인할 수 있다.

```
# SVM_02 예제와 비교해 보기 위한 설정, 1회 검증용
gamma_list = [0.1, 0.3, 0.50625, 0.7, 0.9, 1.1, 1.3, 1.5]
C_list = [12.5]
```



5. 최종 결론

동일한 시드 값(1234)을 사용하여 난수를 생성하였기에 SVM_02와 SVM_03의 정확도는 99.00%로 동일하다. 이는 같은 난수를 생성하기 때문에 결과 값도 동일하게 나타날 수 있었다. 특히 SVM_03에서는 C=12.5 및 gamma=0.3 조합에서 최고 정확도인 99.4%를 기록했다는 것을 확인할 수 있었다.