Python & AI Math 오피스아워

최성철 마스터, 임성빈 마스터

조교 박기훈, 조교 양홍준

1. 기본과제 해설

1.1 Basic Math

1.2 Text Processing I

1.3 Text Processing II

2. 심화과제 해설

2.1 Gradient Descent

2.2 Backpropagation

2.3 Maximum Likelihood Estimation(MLE)

3. Q & A

1.
기본과제

1.1 Basic Math
1.2 Text Processing I
1.3 Text Processing II

1.1 Basic Math

Python에서 기초적인 list와 변수 사용법을 학습하기 위한 과제입니다.

- python 내장 함수로도 많은 기능을 지원하지만, numpy library에 연산을 도와주는 더 많은 함수들이 있으니 둘 다 시도해보시는 것이 좋습니다 !

1.2 Text Processing I

Python에서 String을 다루는 방법을 학습하기 위한 과제입니다.

- split 함수
  - 문자열을 일정한 규칙으로 잘라서 list로 만들어 주는 함수입니다.
  - 문자열.split(sep='구분자', maxsplit=분할횟수)

- join 함수
  - 매개변수로 받은 list에 있는 요소 하나하나를 합쳐서 하나의 문자열로 만들어 주는 함수입니다.
  - '구분자'.join(list) : 각 요소 사이에 구분자를 넣어서 합쳐줍니다.

1.3 Text Processing II

Python에서 String을 다루는 방법을 학습하기 위한 과제입니다.

- digits_to_words
- to_camel_case
  - 컴퓨터 프로그래밍에 사용되는 명명 규칙(변수, 함수 등등) 중 대표적인 2가지 방법
  - underscore(_) string이란 ?
    - 변수 내 단어를 underscore(_)를 사용하여 구분해놓은 것
    - e.g. max_iter, lr_rate, under_score_variable, …
  - camel case란 ?
    - 구분자 없이 대소문자로 구분해놓은 것
    - e.g. camelCaseVariable, mseLoss, …

2.

심화과제

2.1 Gradient Descent

2.2 Backpropagation

2.3 Maximum Likelihood Estimation(MLE)

2.1 Gradient Descent

Gradient Descent를 직접 구현해보기 위한 과제입니다.

- Gradient Descent를 이론적으로 접근해보면 함수식의 미분은 매우 간단하지만, 미분을 코드로 구현하는 것은 하나의 task
  - SymPy library를 사용하는 방법
  - 극한으로 보내는 대신, 그에 아주 가까운 작은 값을 대입해 줌으로써 유사 값을 사용하는 방법

- Linear Regression
  - 주어진 dataset 내에서의 어떤 특징을 Linear하게 표현하는 것
  - 즉, y와 x 간의 선형 관계를 찾는 것

2.1 Gradient Descent

Gradient Descent를 직접 구현해보기 위한 과제입니다.

- Stochastic Gradient Descent
  - Batch 란?
    - 모델을 학습할 때 한 iteration(반복)에 사용되는 sub-dataset
    - Batch size = Batch 하나에 포함되는 data의 개수
  - Gradient Descent가 full-batch라면, SGD는 mini-batch optimization !
    - full-batch를 사용하는 경우, 매 step마다 전체 데이터에 대해 loss function을 계산해야 하므로 너무 많은 계산량을 필요로 하고, 그만큼 시간도 오래 걸림
    - 그러므로 mini-batch를 사용하게 되면, 계산 속도가 훨씬 빨라서 같은 시간 내 훨씬 많은 step을 갈 수 있고, 그만큼 빠르게 optimal point 근처로 갈 수 있음
    - Local minima에 빠지지 않고, 더 좋은 방향으로 수렴할 가능성이 높음

2.2 Backpropagation

RNN의 backpropagation을 직접 구현해보기 위한 과제입니다.

TODO 1 solution:

$$\frac{\partial \xi}{\partial W_x} = \frac{\partial \xi}{\partial y}\frac{\partial y}{\partial S_n}\frac{\partial S_n}{\partial W_x} + \frac{\partial \xi}{\partial y}\frac{\partial y}{\partial S_n}\frac{\partial S_n}{\partial S_{n-1}}\frac{\partial S_{n-1}}{\partial W_x}\cdots = \sum_{k=0}^{n}\frac{\partial \xi}{\partial y}\frac{\partial y}{\partial S_k}\frac{\partial S_k}{\partial W_x} = \sum_{k=0}^{n}\frac{\partial \xi}{\partial S_k}X_k$$

$$\frac{\partial \xi}{\partial W_{rec}} = \frac{\partial \xi}{\partial y}\frac{\partial y}{\partial S_n}\frac{\partial S_n}{\partial W_{rec}} + \frac{\partial \xi}{\partial y}\frac{\partial y}{\partial S_n}\frac{\partial S_n}{\partial S_{n-1}}\frac{\partial S_{n-1}}{\partial W_{rec}}\cdots = \sum_{k=0}^{n}\frac{\partial \xi}{\partial y}\frac{\partial y}{\partial S_k}\frac{\partial S_k}{\partial W_{rec}} = \sum_{k=1}^{n}\frac{\partial \xi}{\partial S_k}S_{k-1}$$

2.2 Backpropagation

Gradient Descent를 직접 구현해보기 위한 과제입니다.

```
'''
TODO 2 SOLUTION:
'''
for k in range(X.shape[1], 0, -1):
    wx_grad += np.sum(
        np.mean(grad_over_time[:,k] * X[:,k-1], axis=0))
    wRec_grad += np.sum(
        np.mean(grad_over_time[:,k] * S[:,k-1]), axis=0)
    grad_over_time[:,k-1] = grad_over_time[:,k] * wRec
```

2.3 Maximum Likelihood Estimation(MLE)

TODO1.(Solution) 로그의 성질을 가지고 위의 수식을 전개하여 식을 간단히 정리해주세요.

(간단히 정리한 수식의 최종적인 형태는 여러가지 모양이 될 수 있습니다.)

$$= \sum_{i=1}^{n} \left[\log\left(\exp(-\frac{(x_i - \mu)^2}{2\sigma^2}) \right) - \log\left(\sigma\sqrt{2\pi}\right) \right]$$

$$= \sum_{i=1}^{n} \left[-\frac{(x_i - \mu)^2}{2\sigma^2} - \log(\sigma) - \log(\sqrt{2\pi}) \right]$$

첫 번째로 $L(\theta|x)$를 모평균 μ에 대해 편미분하면, 아래와 같이 계산되며 미분계수가 0이 되어 최댓값을 갖게 하는 μ를 찾을 수 있습니다.

$$\frac{\partial L(\theta|x)}{\partial \mu} = -\frac{1}{2\sigma^2} \sum_{i=1}^{n} \frac{\partial}{\partial \mu} \left(x_i^2 - 2x_i\mu + \mu^2 \right)$$

2.3 Maximum Likelihood Estimation(MLE)

TODO2.(Solution) 위의 수식을 전개하여 미분계수가 0이 되게 하는 μ 값을 구해주세요.

$$= -\frac{1}{2\sigma^2}\sum_{i=1}^{n}(-2x_i + 2\mu)$$

$$= \frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i - \mu) = \frac{1}{\sigma^2}\left(\sum_{i=1}^{n} x_i - n\mu\right) = 0$$

따라서, 최대우도(또는 최대로그우도)를 만들어주는 모평균의 추정량은 아래와 같습니다.

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i = \bar{x}$$

어디서 많이 본 형태 아닌가요? 모평균은 바로 **표본평균** $\bar{x}$로 추정된다는 것을 알 수 있습니다.

한편, 이번에는 $L(\theta|x)$를 모표준편차 σ에 대해 편미분하면, 아래와 같습니다.

2.3 Maximum Likelihood Estimation(MLE)

TODO3.(Solution) 로그가능도 함수 L을 σ로 편미분한 후, 편미분계수가 0이 되게 하는 σ(또는 σ^2)를 구하세요.

$$\frac{\partial L(\theta|x)}{\partial \sigma} = -\frac{n}{\sigma} - \frac{1}{2}\sum_{i=1}^{n}(x_i-\mu)^2 \frac{\partial}{\partial \sigma}\left(\frac{1}{\sigma^2}\right)$$

$$= -\frac{n}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{n}(x_i-\mu)^2 = 0$$

따라서, 최대우도(또는 최대로그우도)를 만들어주는 모분산의 추정량은 다음과 같다는 것을 알 수 있습니다.

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)^2 = s^2$$

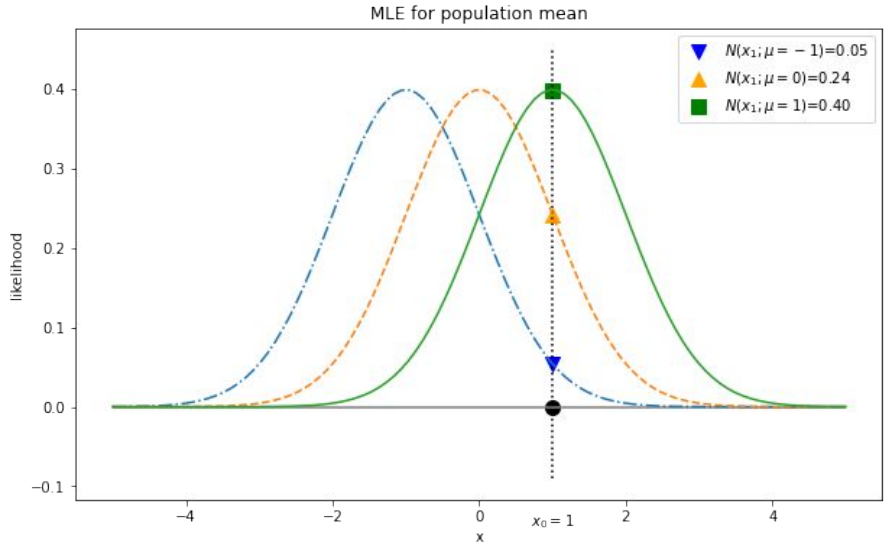
모분산도 모평균과 마찬가지로 표본분산 s^2으로 추정된다는 것을 알 수 있습니다.

결론: 최대가능도 추정법에 의한 정규분포의 기댓값은 **표본평균** 과 같고 분산은 (자유도가 아닌 데이터의 수로 나눈, 편향) **표본분산**과 같습니다.

2.3 Maximum Likelihood Estimation(MLE)

TODO4.(Solution) 다음 코드를 실행하기 위해 빈칸을 완성하세요.

```
# TODO 4-1, 4-2, 4-3 solution
plt.plot(x, sp.stats.norm(loc=-1).pdf(x), ls="-.")
plt.plot(x, sp.stats.norm(loc=0).pdf(x), ls="--")
plt.plot(x, sp.stats.norm(loc=1).pdf(x), ls="-")
```




```
# TODO 4-4, 4-5, 4-6 solution
print('mu=-1: likelihood at x_0=1 is {:.4f}'.format(norm.pdf(x0, -1, 1)))
print('mu=0: likelihood at x_0=1 is {:.4f}'.format(norm.pdf(x0, 0, 1)))
print('mu=1: likelihood at x_0=1 is {:.4f}'.format(norm.pdf(x0, 1, 1)))
```

```
mu=-1: likelihood at x_0=1 is 0.0540
mu=0: likelihood at x_0=1 is 0.2420
mu=1: likelihood at x_0=1 is 0.3989
```

2.3 Maximum Likelihood Estimation(MLE)

TODO5.(Solution) 그래프를 그리고 관찰한 사실을 바탕으로 다음의 빈칸의 결과를 기입해주세요.

그려진 그래프의 관찰을 통해 다음의 사실을 확인할 수 있습니다.

- $N(x; \mu = -1)$이라는 확률분포에서 $x_0 = 1$이 나올 가능도(확률밀도)는 **0.05**이다.
- $N(x; \mu = 0)$이라는 확률분포에서 $x_0 = 1$이 나올 가능도(확률밀도)는 **0.24**이다.
- $N(x; \mu = 1)$이라는 확률분포에서 $x_0 = 1$이 나올 가능도(확률밀도)는 **0.40**이다.

어떤 모평균을 갖는 확률분포를 고르는 것이 합리적일까요? 당연히 가장 큰 가능도를 가진 확률분포를 선택해야 합니다. 그림에서 볼 수 있듯이 μ **=1** 일 때(녹색 그래프) 가장 큰 값의 Likelihood를 갖게 되며 최대가능도 추정에 의해 모평균의 추정값은 $\hat{\theta}_{\text{MLE}} = \arg\max_\theta L(\theta)$ **=1** 입니다.

3.

Q & A

End of Document

Thank You.