1. 데이터 분석 이해하기

전형적인 데이터 분석과정은 일반적인 과학적 발견의 절차와 같다. 데이터 과학에서는 대답해야할 질문과 적용해야 할 분석 방법에서 발견이 시작된다. 가장 간단한 형태의 분석 방법은 기술(descriptive) 통계로 데이터를 취합해 시각화한 형태로 표현한다. 주어진 데이터의 샘플의 크기가 작고 통계에 기반한 추정에 적합하다. 예측(prediction)을 위한 데이터 분석은 과거의 자료를 사용하여 미래를 예측하며 한다. 인과(causal) 분석은 서로에게 영향을 미치는 변수를 식별한다. 역학(mechanistic) 데이터 분석은 변수가 다른 변수에 정확히 어떤 영향을 주는지 탐구한다.

데이터 분석 품질은 얼마나 좋은 데이터를 사용하느냐에 의해 결정된다. 데이터분석을 위해 필요한 자료인 데이터셋(data set)은 웹 또는 공개 API 을 통해 획득하고 데이터 정제 도구와 통계적 지식을 활용해서 데이터셋을 정규화해야 한다.

데이터를 정제한 후에는 기술 통계분석과 탐색적분석을 수행한다. 이 단계에서 결과물은 산포도(scatter plot) , 히스토그램, 통계적 요약이다.

정제된 데이터를 사용하여 데이터의 모델을 학습하여 예측을 할 수 있다. 이 과정에서 예측 정확도를 평가하여 정확도가 낮은 경우 데이터 수집, 정재 및 재학습을 반복한다.

데이터셋의 품질은 통계학자와 프로그래머에 의해 결정하지 않으며 도메인 전문가에 의해 판단하게 된다.

여기서는 데이터 분석의 준비 단계를 중심으로 설명하며 준비 단계에서는 데이터 수집, 전처리, 정리 및 분류의 과정을 수행한다. 다른 단계에 비해 정형화되지 않아 다양한 창의적 접근이 가능하다.

데이터 수집은 다양한 출처에서 입력 데이터가 포함된 아티팩트(artifact) 를 확득하고, 아티팩트에서 데이터를 추출하며, 추출한 데이터를 추가적인 처리에 적합한 형태로 변환하는 과정을 포함한다.

데이터를 얻는데 사용하는 출처는 인터넷, 데이터베이스, 파일이다. 데이터의 포멧에 따라 자연어로 된 비정형 텍스트, 반정형 및 정형데이터로 나눈다. 반정형 데이터는 HTML, XML, JSON 과 같이 자료의 표현을 위한 규칙은 있으나 내용이 정형적이 않은 경우이며 정형데이터는 테이블 형 데이터이다. CSV 파일, 데이터베이스에 저장되어 유지한다.

추출된 데이터는 추가적인 목적과 특성에 따라 파이썬에 내장된 데이터 구조(리스트, 딕션너리), 특수한 연산(numpy 배열, pandas데이터프레임)을 지원하는 고급데이터 구조로 표현한다.

배경 지식이해하기 위해 인공지능 시스템, 머신러닝 및 파이썬 언어에 대한 설명한다.

1. 인공지능 시스템이란

(기존시스템)설정화면에서 원하는 온도를 입력하는 행동으로 에어컨 동작 조건을 설정한다.

(인공지능시스템)기기 조작에 익숙하지 않은 사람까지 고려해서 에어컨의 설정온도를 자동으로 조정하게 만든다.

인공지능 시스템을 구현하기 위해서는 에어컨을 켰을 때의 온도를 저장한 데이터가 있어야 한다. 즉, 데이터를 살펴보면 실내 온도가 일정 동도 보다 높을 때 에어컨을 켜는 것을 확인하 수 있다.

이와 같이 사람의 행동에서 기기가 적정 온도를 판단하면 설정 화면이 필요 없게 된다. 인공지능 시스템을 개발하기 위해서는 이와 같이 측정한 온도 데이터를 이용해 컴퓨터가 적정 온도를 학습하는 과정이 필요하다.

이전 시스템(기존 시스템)은 개발하는 사람이 데이터에서 규칙을 생각해 프로그램을 개발하였으나 인공지능은 컴퓨터가 데이터에서 자동으로 규칙을 만들어 처리까지 하는 프로그램을 만드는 것이다.

1. 데이터 기반 학습

24도 이상에서 에어컨을 켰다는 규칙을 정했다고 가정하자. 최근 10일 동안 에어컨을 켤 것으로 예측(24도 이상) 하는 날이 5일, 실제로 에어컨을 켠 날은 3일이다. 반대로 에어컨을 켜지 않을 것으로 예측(24도 미만) 하는 날은 5일인데 실제 특정일에는 에어컨을 켰다. 이 경우 실제 예측 데이터와 결과 데이터가 일치할 떄 정확률은 (3+4)/10 = 0.7 이므로 70% 이다.

규칙을 조정하여 기준 설정 온도를 변경시켜 정확률을 높일 수 이다. 설정 온도를 26도 이상으로 정하면 정확률은 8/10 = 0.8 이며, 22도 이상으로 에어컨을 켜는 규칙의 정확률은 7 / 10 = 0.7 이므로 70% 이다.

학습 데이터와 테스트 데이터를 적절히 정하는 방법으로 짝수 날의 데이터를 학습데이터로 홀 수 날의 데이터를 테스트 데이터로 설정하여 학습하면 가장 적절한 온도를 찾을 수 있다. 데이터를 바꿔 정확도를 확인하는 것을 교차 검증이라고 한다. 또한 얻은 데이터가 많을 수록 정확도가 높을 것이다.

최근 인공지능 연구와 개발은 단순한 연구에서 벗어나 다수의 기업이 인공지능 활용을 위해 참여하고 있다. 다음은 인공지능 활용이 확대되게 된 배경을 설명한 것이다.

* 소스코드나 API 가 공개되어 다른 기업이나 대학 등에서 연구 성과를 이용할 수 있다.
* 여러 대의 컴퓨터를 활용하는 병렬 처리를 위한 클라우드 구성이 용이하다
* 많은 데이터를 모아 처리할 수 있다
* GPU 활용 등으로 컴퓨터의 처리 속도가 향상되었다.

1. 머신 러닝과 데이터 처리(분석)

사람은 문제를 해결하기 위해 과거의 경험에서 얻는 직관으로 해결하는 경우가 많다. 즉, 경험에서 규칙을 발견해 문제 해결방법으로 적용하는 것이다.

이러한 과정을 컴퓨터에서 처리한다면 데이터(과거의 경험) 에서 규칙을 이끌어내는 학습을 거쳐 문제 해결에 적용하는 것이다. 여기서 데이터를 바탕으로 컴퓨터가 규칙을 이끌어 내는 과정을 기계학습(머신 러닝)이다.

(생각하기) 지금까지 컴퓨터 연산은 사람이 입력한 데이터를 컴퓨터 안 함수에 전달해 처리한 후 결과를 출력한다. 그러나 머신러닝의 연산을 컴퓨터가 처리해야 할 함수를 직접 추측해 정의하는 것이다.

(변화 이해하기) 대량의 데이터에서 규칙을 조사하는 작업은 반복할 때 실수할 가능성이 높다. 머신러닝이 주목받는 이유는 사람이 직접 처리할 수 없을 정도의 데이터(빅데이터)가 모였고 전세계에서 데이터를 사용할 수 있는 데이터 수집 및 저장이 쉽게 되었기 때문이다.

머신러닝은 데이터양이 적은 상황에서는 정확한 결과를 얻기 어려울 때가 많다. 이러한 경우 여전히 사람의 판단으로 문제를 해결하는 것이 효율적이다.

머신러닝의 핵심은 “해결하고 싶은 문제를 어떻게 수치화하는가” 이다. 컴퓨터는 데이터를 수치화하지 않으면 학습할 수 없고, 효율적으로 수치화하지 않으면 학습결과의 정확도에 큰 영향을 받는다.

1. 머신러닝에 사용하는 데이터

머신러닝에 사용하는 데이터는 일정 수준 이상의 양이 있어야 한다. 그러나 무조건 데이터 양이 많다고 좋을 것이 아니며 데이터의 품질이 보장되어야 한다. 머신러닝을 공부할 때 필요한 데이터를 준비하는 것은 쉬운 작업이 아니다. 머신러닝은 데이터를 준비하기가 번거로우며 데이터 정재 작업과 변환작업이 필수적으로 요구된다.

대표적으로 머신러닝에서 사용되는 붓꽃 데이터 셋은 붓꽃 데이터 150개를 세가지 품종으로 분류한다.

1. 머신러닝을 위한 프로그래밍 언어

프로그램을 효율적으로 만들려면 다양한 프로그래밍 언어의 장단점을 파악해 만들려는 시스템이나 서비스에 맞는 최적의 프로그래밍 언어를 선택해야 한다. C 나 C++ 같은 프로그래밍 언어는 빠른 처리 속도가 장점이므로 운영체제나 게임개발 등에 사용한다. 머신러닝에서도 빠른 처리 속도가 필요한 라이브러리는 C 언어로 구현한다. 우리는 머신러닝 라이브러리를 개발하는 것을 목적으로 하지 않으므로 C 나 C++ 같은 프로그래밍 언어를 사용하지 않는다.

프로그래밍 언어를 선택할 때는 프로그램을 실제로 실행할 환경도 고려해야 한다. 안드로이드 앱을 개발할 때는 자바(Java) 나 코틀린(Kotlin), iOS 앱을 개발할 때는 오브젝트-C 나 스위프트(Swift) 를 사용하다. 머신러닝은 통계와 밀접한 관련이 있으며, 통계분야에서 주로 사용하는 프로그래밍 언어는 파이썬, R, 매트랩(Matlab) 등이 있다. 머신러닝, 딥러닝 프로그램 개발에는 파이썬을 주고 사용한다. 파이썬은 머신러닝이나 딥러닝 프로그래밍에 필요한 라이브러리가 갖추어져 있어 원하는 기능 구현에 집중 할 수 있기 때문이다.

여기서도 파이썬을 사용한다. 인공지능 개발이나 데이터 분석에 파이썬을 활용하는 개발자가 증가하고 있으며 관련 참고 자료를 쉽게 찾을 수 있기 때문이다.

1. 파이썬의 특징

프로그램은 크게 구현과 실행이라는 부분으로 나눌 수 있다. 구현은 프로그래밍 언어의 문법 등에 따라 코드를 작성하는 것이다. 실행은 작성한 코드를 실행하는 것이다.

실행에는 보통 세가지 요소가 사용된다.

* 컴파일러와 인터프리터: 코드를 해석해서 실행하도록 변환하는 프로그램이다.
* 라이브러리: 프로그래밍할 때 자주 사용하는 기능을 모아 미리 만들어 놓은 코드이다.
* 실행환경: 프로그램을 실제로 동작시키는 환경으로 하드웨어 사항이나 운영체제등이 포함된다.

컴파일러와 인터프리터는 역할은 같지만 중간과정이 차이점이 있다. 인터프리터는 코드를 해석한 후 별도의 변환 과정없이 수행된다. 즉, 변환하는 실행파일이 없다. 코드 한줄을 해석한 후 바로 실행한다. 파이썬은 인터프리터 언어로 분류되며 스크립트 언어로도 분류 한다. 같은 동작을 실행하는 C 나 자바 프로그래밍 언어 보다 간결하게 작성할 수 있으므로 코드 길이가 짧다는 특징이 있다.

파이썬은 1991년에 처음 발표 했으며, 현재 2.x 와 3.x 가 모두 사용되고 있다. 이들 버젼은 호환성을 유지하지 않고 있다. 최근 2.x 에 대한 지원이 종료됨으로 써 새롭게 파이썬 언어를 사용하는 경우 3.x 사용을 권고한다.

1. 스크립트 파일 사용하기

파이썬 코드는 대화형 모드에서 입력해 실행한다. 그러나 파이썬은 코드를 스크립트 파일로 저장해 실행할 수 있다. 실제 많은 프로그래밍 언어는 대화형 모드를 지원하지 않고 파일에 코드를 저장하고 실행하는 것이 일반적이다.

스크립트 파일를 만들어 코드를 입력해 두면 해당 파일로 같은 작업을 다시 실행할 수 있다. 파이썬 스크립트 파일은 . py 라는 확장자를 사용한다.

스크립트 파일 역시 컴파일하지 않고 해석방식으로 처리한다.

1. 결과보고서 구조

데이터 분석 과정은 보고서 형태로 작성한다. 보고서는 다음 항목으로 구성된다.

* 요약
* 서론
* 데이터 수집과 처리에 사용한 방법
* 분석결과 (중간결과의 자료 또는 표 또는 차트는 부록에 작성함)
* 결론
* 부록

부록에는 부차적인 결과와 차트 뿐만 이나라 데이터를 처리하는데 사용하는 코드를 포함한다.