**탐색적 데이터 분석**

**(Exploratory Data Analysis, EDA)**

작성자: 윤경애

1. 정의

* 원본 데이터를 가지고 데이터를 탐색하고, 데이터의 특징과 구조로부터 얻은 정보를 바탕으로 통계모형을 만드는 분석방법

1. 목표

* 관측된 현상의 원인에 대한 가설을 제시하고, 적절한 통계 도구 및 기법의 선택을 위한 가이드가 되며, 통계 분석의 기초가 될 가정을 평가하고 추가 자료 수집을 위한 기반을 제공함

1. 절차

* 데이터 수집 -> 시각화 탐색 -> 패턴 도출 -> 인사이트 발견
* 아래와 같은 방법을 반복하여 데이터를 이해 및 탐구하는 과정
* 데이터에 대한 질문 & 문제 만들기
* 데이터를 시각화하고, 변환하고, 모델링함으로써 질문 & 문제에 대해 답하기
* 찾는 과정에서 배운 것들을 토대로 다시 질문을 다듬고 또 다른 질문 & 문제 만들기
* **이러한 과정을 기반으로 데이터에서 흥미있는 패턴이 발견될 때까지, 더 찾는 것이 불가능하다고 판단될 때까지** 도표, 그래프 등의 시각화, 요약 통계를 이용하여 **전체적인 데이터를 살펴보고 개별 속성의 값을 관찰**함. 데이터에서 발견되는 **이상치를 찾아내 전체 데이터 패턴에 끼치는 영향을 관찰**하고, **속성간의 관계에서 패턴을 발견**함
  1. 전체적인 데이터 살펴보기
* 데이터 항목의 개수, 속성 목록, NAN 값, 각 속성이 가지는 데이터형 등 확인
* 데이터 가공과정에서 데이터 오류나 누락이 없는지 head, tail을 확인
* 데이터를 구성하는 각 속성값이 예측한 범위와 분포를 갖는지 확인
  1. 이상치(outlier) 분석
* 개별 데이터를 관찰하여 전체적인 추세와 특이사항을 관찰
* 데이터가 많다고 특정 부분만 보게되면 이상치가 다른 부분에서 나타날 수도 있으므로 앞, 뒤, 무작위로 표본을 추출해서 관찰해야 함. 이상치들은 작은 크기의 표본에서는 나타나지 않을 수도 있음
* 통계지표를 활용; 데이터의 중심을 알기 위해서는 평균, 중앙값, 최빈값 사용, 데이터 분산도를 알기 위해서는 범위, 분산 사용
* 시각화 활용; 시각화를 통해 데이터의 개별 속성에 어떤 통계 지표가 적절한지 결정, 시각화 방법에는 histogram, scatter plot, box plot, 시계열 차트, K-means 기법, stochastic based detection, deviation based method, distance based detection 기법
  1. 속성 간의 관계 분석
* 서로 의미있는 상관관계를 갖는 속성의 조합 찾기
* 분석의 대상이 되는 속성의 종류에 따라서 분석 방법도 달라져야 함
* 범주형 변수(명목형/순서형 데이터) + 수치형 변수(연속형/이산형 데이터)
* 수치형 변수-수치형 변수 간의 관계 분석: 상관계수를 통해 두 속성간의 연관성을 나타냄. Heatmap 또는 scatter plot를 이용하여 시각화 가능
* 수치형 변수-범주형 변수 간의 관계 분석: 카테고리별 통계치를 범주형으로 나누어서 관찰할 수 있고, box plost, PCA plot 등으로 시각화 가능
* 범주형 변수-범주형 변수 간의 관계 분석: 각 속성값의 쌍에 해당하는 값의 개수, 분포를 관찰 가능하고 pie chart, mosaic plot 등으로 시각화 가능

1. 특징

기술통계 기반

데이터의 분포와 값을 다양한 각도에서 관찰하며 데이터가 표현하는 현상을 이해할 수 있도록 도와주고 데이터를 다양한 기준에서 살펴보는 과정을 통해 문제 정의 단계에서 발견하지 못한 다양한 패턴을 발견하고 이를 바탕으로 기존의 가설을 수정하거나 새로운 가설을 추가할 수 있도록 한다.

데이터 탐색 결과는 통계적 추론이나 예측 모델 구축 시에도 사용되므로 중요한 데이터 분석 단계임

1. 예시

5.1 전체적인 데이터 살펴보기

|  |
| --- |
| # 데이터 불러오기  df = pd.read\_csv(‘xxx/xxx/xxx.csv’)  # 데이터 항목의 개수와 타입 알아보기  print(df.shape) # 데이터 행 열 개수  print(df.types) # 데이터 타입 출력  # 데이터의 head, tail 살펴보기  df.head()  df.tail()  # 중복값 확인  df[df.duplicated(keep=False)] # 중복된 열 출력  df = df.drop\_duplicates() # 중복된 열 제거  df.shape # 제거된 열 확인  # nan 값 확인 및 처리  df.isna() # nan 값이 있는지 출력(true, false 형태로 출력)  df.isna().sum() #열별 nan 값 개수 출력  df = df.dropna() # nan값을 제거, fillna() 함수로 nan 값을 치환할 수도 있음  df.shape # 제거된 열 확인 |

5.2 이상치 분석

|  |
| --- |
| # 컬럼별 요약 통계 지표 확인  df.describe() # 컬럼별 요약 통계(개수, 평균, 표준편차, 최소값, 최대값, 4분위수)  # 데이터 분포 확인  # histogram: 어느 부분에 자료가 집중되어 있는지와 이상치를 검토할 수 있고 데이터의 좌우 대칭성을 설명할 수 있음, cf. 계급 폭을 다르게 설정하면 해석이 달라질수 있음  plt.hist(df[‘sepal\_width’], bins=10)  sns.histplot(df[‘sepal\_width], bins=10)  # scatter plot: 두 변수 간의 방향, 트렌드, 밀집도와 이상치를 확인 가능, cf. 명목형 변수의 군집별 색을 다르게 하여 구분 가능  scatter_plot_python.jpg  #box plot: 최소값, 최대값, 4분위수와 중앙값, 이상치 확인 cf. describe()  data = df.iloc[:, :-1]  data.boxplot() |

5.3 속성간 관계 분석

|  |
| --- |
| # heatmap  plt.figure(figsize=(7,4))  sns.heatmap(df.corr(), annot=True) # 상관계수를 heatmap으로 시각화  # pair plot: 데이터의 모든 변수(컬럼)의 상관관계를 histogram과 scatter plot으로 출력, 전체 데이터의 상관관계를 한 눈에 볼 수 있음  plt.figure(figsize=(10,8))  sns.pairplot(df)  # PCA plot: 여러 변수의 변량을 주성분으로 요약, 축소하는 방법으로 먼저 scatter plot을 이용하여 주성분의 수를 정하고 이를 바탕으로 PCA plot을 그려 분포와 주성분간의 관계를 확인, 각 주성분이 차지하는 분산의 누적비율을 계산해서 각 주성분이 전체 분산 중 얼마만큼 설명해 주는지 알 수 있음 |

참고자료: <https://post-blog.insilicogen.com/blog/361>