SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

삼성 청년 SW 아카데미

데이터분석 이론 및 실습

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

9월12일 실습 및 과제 알고리즘

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

실습 알고리즘

실습1. 기본 시각화 - 서울시 데이터 분석

- 1-1. 데이터 로드 및 요일별, 시간대별 이용객 수 집계

1st. 데이터를 불러오기 위해 pd.read_csv()를 사용하여 파일을 로드합니다

2nd. 요일별 이용객 수를 집계하기 위해 groupby('DayOfWeek')와 sum()을 사용하여 각 요일의 총 이용객 수를 계산합니다

3rd. 시간대별 이용객 수를 집계하기 위해 groupby('Time')와 sum()을 사용하여 각 시간대의 총 이용객 수를 계산합니다

• 1-2. 요일별 이용객 수 가로 막대 그래프 시각화

1st. 요일별 이용객 수를 시각화하기 위해 plt.barh()를 사용하여 가로 막대 그래프를 그립니다

2nd. 그래프의 가독성을 높이기 위해 데이터를 sort_values()로 정렬하여 출력합니다

• 1-3. 시간대별 이용객 수 꺾은선 그래프 시각화

1st. 시간대별 이용객 수의 변화를 시각화하기 위해 plt.plot()을 사용하여 꺾은선 그래프를 그립니다

2nd. 꺾은선 그래프에 마커와 그리드를 추가하여 시각적 요소를 강화하고 가독성을 높입니다

실습2. Seaborn을 활용한 서울시 데이터 시각화 심화

• 2-1. 키 분포 히스토그램 및 KDE 그래프 시각화

1st. 설문조사 데이터를 불러오기 위해 pd.read_csv()를 사용하여 파일을 로드합니다

2nd. 참가자들의 키 분포를 시각화하기 위해 sns.histplot()을 사용하여 히스토그램과 커널 밀도 추정(KDE)을 그립니다

• 2-2. 성별에 따른 키 분포를 바이올린 플롯으로 비교

1st. 성별에 따른 키 분포 차이를 시각적으로 비교하기 위해 sns.violinplot()을 사용하여 바이올린 플롯을 생성합니다

2nd. 그래프의 축 레이블과 제목을 설정하여 시각적 정보를 명확하게 전달합니다

• 2-3. Seaborn 스타일 옵션 적용

1st. Seaborn의 스타일 옵션을 적용하기 위해 sns.set(style='whitegrid')를 사용하여 그래프의 시각적 품질을 향상시킵니다

2nd. 스타일이 적용된 바이올린 플롯을 다시 생성하여 시각적 효과를 강화합니다

- 3-1. 세계 인구 및 GDP 데이터를 불러와 데이터프레임으로 병합

1st. 세계 인구 데이터를 불러오기 위해 pd.read_csv()를 사용하여 population_data.csv 파일을 로드합니다

2nd. 세계 GDP 데이터를 불러오기 위해 pd.read_csv()를 사용하여 gdp_data.csv 파일을 로드합니다

3rd. 인구와 GDP 데이터를 국가와 연도를 기준으로 병합하기 위해 pd.merge()를 사용합니다

- 3-2. Plotly를 사용하여 인구와 GDP 간의 상관관계를 점 그래프로 시각화

1st. 병합된 데이터프레임을 사용하여 인구와 GDP의 상관관계를 시각화하기 위해 px.scatter() 함수를 사용합니다

2nd. 점의 크기와 색상을 통해 국가별 인구와 GDP를 표현하여 데이터 간의 관계를 시각적으로 파악할 수 있도록 합니다

• 3-3. Hover 기능 추가

1st. Plotly의 hover_name 옵션을 사용하여 그래프에 마우스를 올렸을 때 국가 이름과 인구, GDP 등 상세 정보가 표시되도록 설정합니다

- 3-4. 슬라이더 기능 추가

1st. 연도별 변화를 확인할 수 있도록 animation_frame 옵션을 사용하여 슬라이더를 추가합니다

2nd. 슬라이더를 통해 특정 연도의 데이터를 선택하고, 시간에 따른 인구와 GDP의 변화를 인터랙티브하게 탐색할 수 있게 합니다

• 4-1. Matplotlib을 사용하여 꺾은선 그래프 생성

1st. 데이터를 불러오기 위해 pd.read_csv()를 사용하여 time_series_data.csv 파일을 로드합니다

2nd. 연도별 특정 지표의 변화를 시각화하기 위해 plt.plot() 함수를 사용하여 꺾은선 그래프를 그립니다

3rd. 그래프의 제목, 축 레이블, 그리드를 추가하여 시각적 요소를 강화하고 가독성을 높입니다

• 4-2. Seaborn을 사용하여 꺾은선 그래프 생성

1st. Seaborn의 sns.lineplot() 함수를 사용하여 동일한 데이터를 시각화합니다

2nd. Seaborn의 기본 스타일과 색상 팔레트를 활용하여 시각적으로 일관된 그래프를 생성합니다

• 4-3. Plotly를 사용하여 꺾은선 그래프 생성

1st. Plotly의 px.line() 함수를 사용하여 상호작용이 가능한 꺾은선 그래프를 생성합니다

2nd. 그래프에 마우스를 올리면 데이터 포인트의 상세 정보를 확인할 수 있어, 데이터 탐색과 프레젠테이션에 유용합니다

실습5. Plotly를 활용한 서울시 데이터 고급 3D 시각화

- 5-1. 세계 각국의 인구, GDP, 평균 수명 데이터를 로드하고 확인

1st. 데이터를 불러오기 위해 pd.read_csv()를 사용하여 world_data.csv 파일을 로드합니다

2nd. 데이터를 확인하기 위해 head() 함수를 사용하여 데이터의 첫 몇 줄을 출력하고 구조와 내용을 파악합니다

실습5. Plotly를 활용한 서울시 데이터 고급 3D 시각화

• 5-2. Plotly를 사용하여 3D 산점도 작성

1st. 3D 산점도를 작성하기 위해 px.scatter_3d() 함수를 사용하고, 인구, GDP, 평균 수명을 각각 x, y, z 축에 할당합니다

2nd. 각 국가의 경제적 및 사회적 특성을 시각적으로 비교할 수 있도록 다양한 색상과 크기로 점을 표시합니다

• 5-3. Hover 기능 추가 및 국가명 표시

1st. Hover 기능을 추가하기 위해 hover_name 옵션을 사용하여 각 점에 마우스를 올렸을 때 국가 이름과 추가 정보를 표시합니다

2nd. 이를 통해 사용자가 데이터 포인트에 대한 세부 정보를 쉽게 확인할 수 있도록 설정합니다

• 5-4. 필터 기능 추가

1st. 연도별 데이터를 필터링할 수 있도록 animation_frame 옵션을 사용하여 슬라이더 기능을 추가합니다

2nd. 슬라이더를 통해 특정 연도의 데이터를 선택하고, 시간에 따른 국가별 인구, GDP, 평균 수명의 변화를 시각적으로 탐색할 수 있도록 합니다

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

과제 알고리즘

과제1. Matplotlib를 사용한 복합 시각화

- 1-1. category_data.csv 파일을 로드하고 데이터를 확인

1st. 데이터를 불러오기 위해 pd.read_csv()를 사용하여 category_data.csv 파일을 로드합니다

2nd. 데이터를 확인하기 위해 head()와 describe()를 사용하여 데이터의 구조와 각 열의 특성을 파악합니다

과제1. Matplotlib를 사용한 복합 시각화

• 1-2. Matplotlib를 사용하여 서브플롯에 다양한 그래프 배치

1st. plt.subplots() 함수를 사용하여 1행 3열의 서브플롯을 생성하고, 각 서브플롯에 다른 유형의 그래프를 추가합니다

2nd. 가로 막대 그래프: barh() 함수를 사용하여 카테고리별 수치 값을 표현하고, 비교할 수 있도록 시각화합니다

3rd. 파이 차트: pie() 함수를 사용하여 서브 카테고리별 수치 값의 비율을 표현하고, 각 서브 카테고리의 상대적인 크기를 시각적으로 나타냅니다

4th. 꺾은선 그래프: plot() 함수를 사용하여 카테고리별 수치 값의 변화 추이를 시각화하고, 데이터의 흐름과 경향을 파악합니다

• 1-3. 서브플롯 레이아웃 조정

1st. plt.tight_layout() 함수를 사용하여 서브플롯 간의 간격을 조정하고, 각 그래프가 겹치지 않도록 배치하여 가독성을 높입니다

과제2. Seaborn을 활용한 고급 통계 시각화

- ## 2-1. statistical_data.csv 파일을 로드하고 데이터를 확인

1st. 데이터를 불러오기 위해 pd.read_csv()를 사용하여 statistical_data.csv 파일을 로드합니다

2nd. 데이터를 확인하기 위해 head()와 describe()를 사용하여 각 변수의 분포와 관계를 이해합니다

과제2. Seaborn을 활용한 고급 통계 시각화

- ## 2-2. Seaborn을 사용하여 통계적 시각화를 서브플롯으로 배치

1st. plt.subplots() 함수를 사용하여 서브플롯을 생성하고, 각 서브플롯에 다양한 통계적 시각화를 구현합니다

2nd. 산점도 및 회귀선 그래프: sns.regplot()을 사용하여 두 변수 간의 상관 관계와 회귀선을 시각적으로 나타냅니다

3rd. 바이올린 플롯과 커널 밀도 추정(KDE) 플롯: sns.violinplot()과 sns.kdeplot()을 사용하여 그룹별 변수의 분포와 밀도를 시각적으로 표현합니다

• 2-3. 레이아웃 조정

1st. plt.tight_layout() 함수를 사용하여 서브플롯 간의 간격을 조정하고, 그래프가 겹치지 않도록 배치하여 전체 시각화의 가독성을 높입니다

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

내일 방송에서 만나요!

삼성 청년 SW 아카데미