

INTERACTIVE DASHBOARD

Covid-19 발생 후 업종별 데이터를 중심으로

발표자 : 박선희





Contents

프로젝트 소개

프로젝트 배경

프로젝트 목적

프로젝트 프로세스

프로젝트 일정

프로젝트 단계별 내용

데이터 EDA

Dashboard - 지역/업종별 매출 분석기능

Dashboard - 맞춤형 매출 변화 분석기능

프로젝트 결과

추가분석

자체평가

보완점 & 목표

참고자료, 공공데이터

1. 프로젝트 소개



[한국정보문화원] 코로나19 장기화 이후 국민 문화생활 변화 양상은?

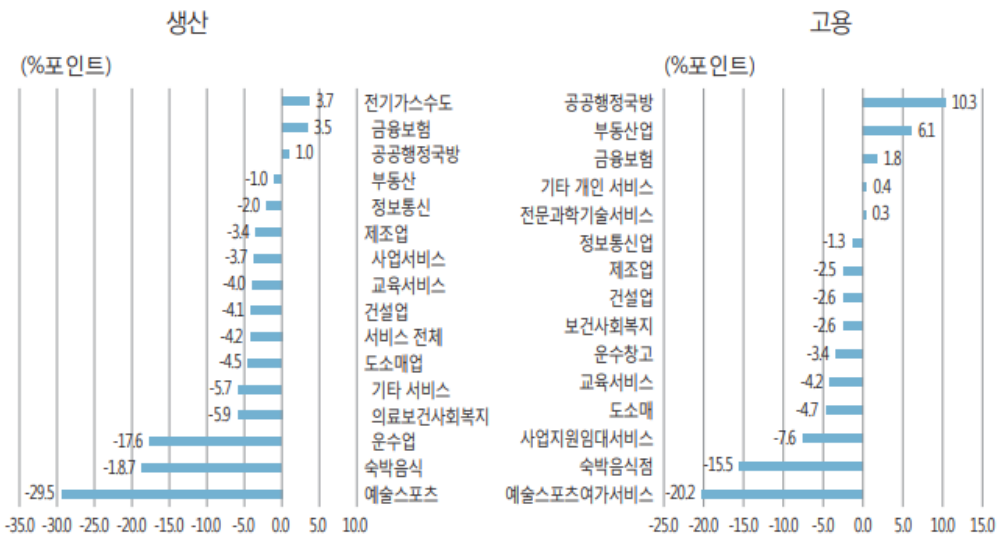
문수인 기자 | 2021.09.24 10:30 | 댓글 0

가 가

상반기(1월~6월) 해외영화 전년도 대비 60.40% 증가
반면 한국영화 매출액 -76.43%로 감소
중극장 공연 매출액도 -13.39%로 감소

[문화뉴스 문수인 기자] 주요 포털 사이트 다음과 네이버 뉴스의 공연·전시 및 영화 뉴스 키워드 분석 결과, '코로나', '금지', '취소', '연기' 등의 부정적 언급이 많았던 2020년 상반기와 달리 2021년 상반기는 코로나 관련 언급이 줄고 '진행', '지원', '백신', '온라인' 등 문화생활 유지 및 회복 관련 키워드가 나타났다.

〈그림 3〉 주요 산업의 생산과 고용에 미친 영향



자료 : 한국은행 ECOS, 고용노동부, 사업체 노동력 조사.

코로나19 장기화로 인해 많은 산업 부문과 경제, 문화 등 많은 것이 변화
항상 뉴스 등 소식으로만 접했지, 실제로 코로나19의 영향 및 변화에 대해 깊이 생각해 본 적이 거의 없었음

경제학도로서 코로나로 인한 국내 산업 구조 변화를 이해하고 싶었기에
데이터 시각화를 통해 분석하고 싶었기에 프로젝트를 수행하게 됨

코로나 영향력 분석

코로나 직후와 대비해 장기화로 접어든
현재의 데이터와 비교해 산업 구조 부문
에서의 변화를 분석하고자 함

산업 구조
변화 파악

올바른 시각화 학습

사실의 왜곡이 없으면서도
쉽고 명확하게 시각화 결과물을
이해할 수 있도록 하는 시각화를
수행하고자 함

정교한
데이터 시각화

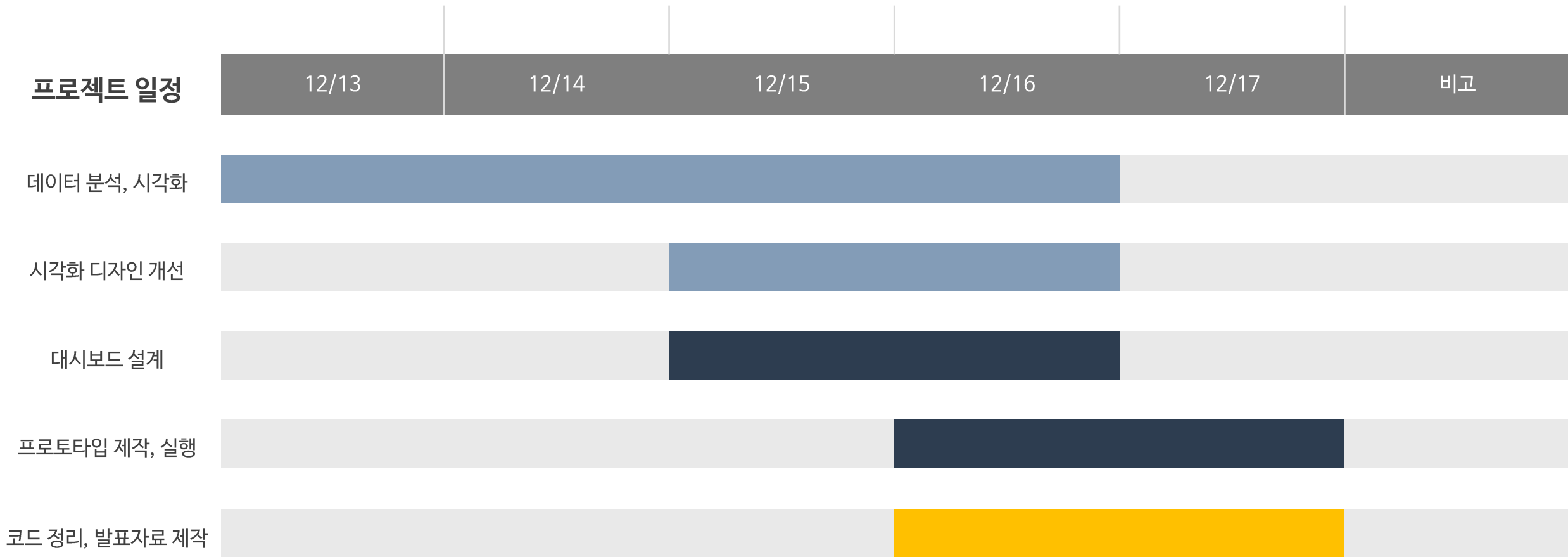
Interactive
Dashboard

대시보드 설계 및 사용

데이터 분석을 편하고 효율적으로
도와주는 대시보드를 설계, 사용해보면서
통찰력과 사용자 니즈를 얻고자 함

2. 프로젝트 프로세스

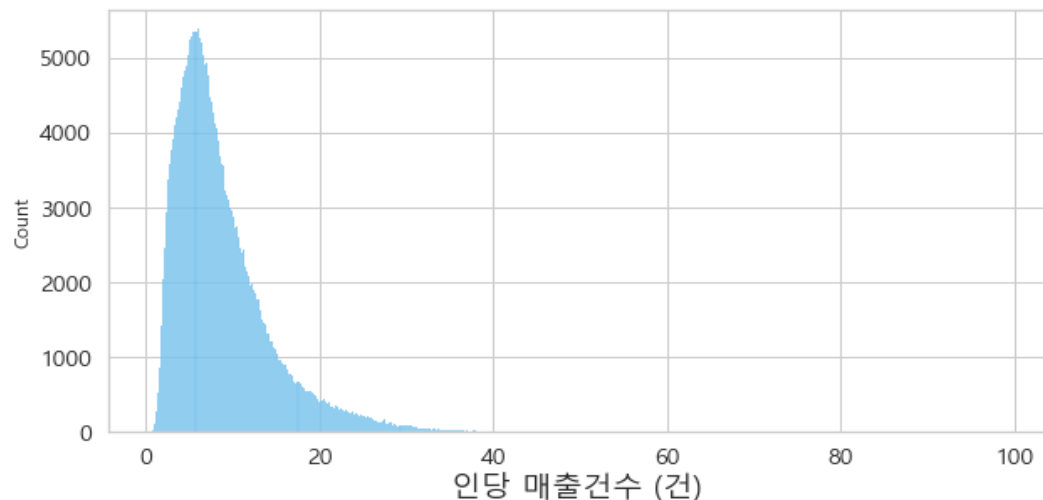
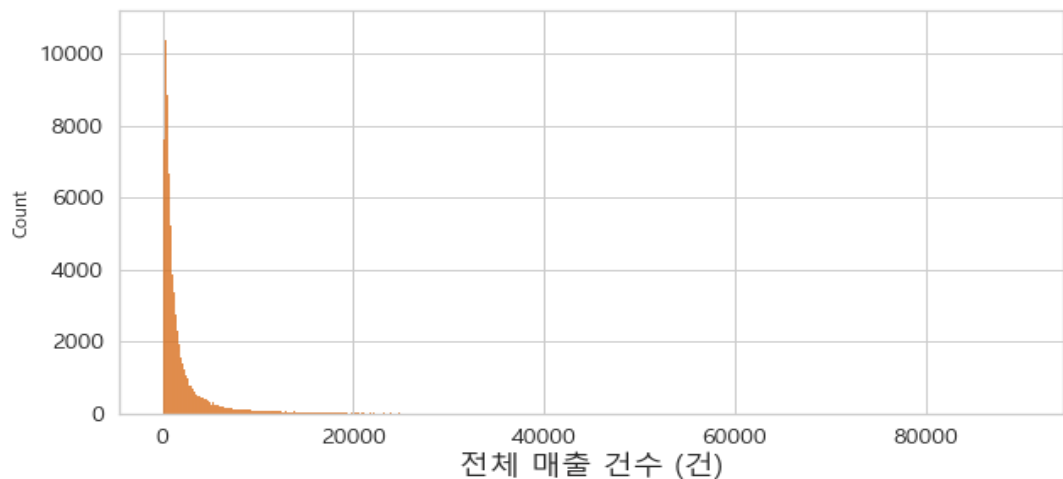




3. 프로젝트 단계별 내용



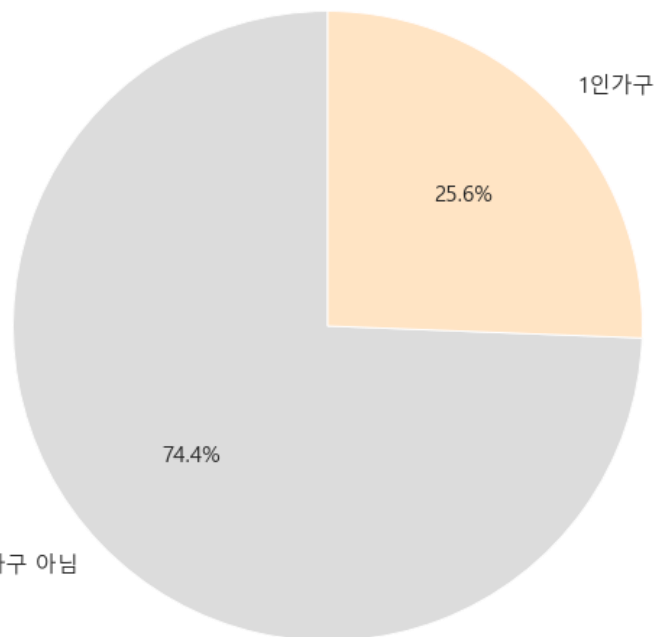
1. 데이터 특징



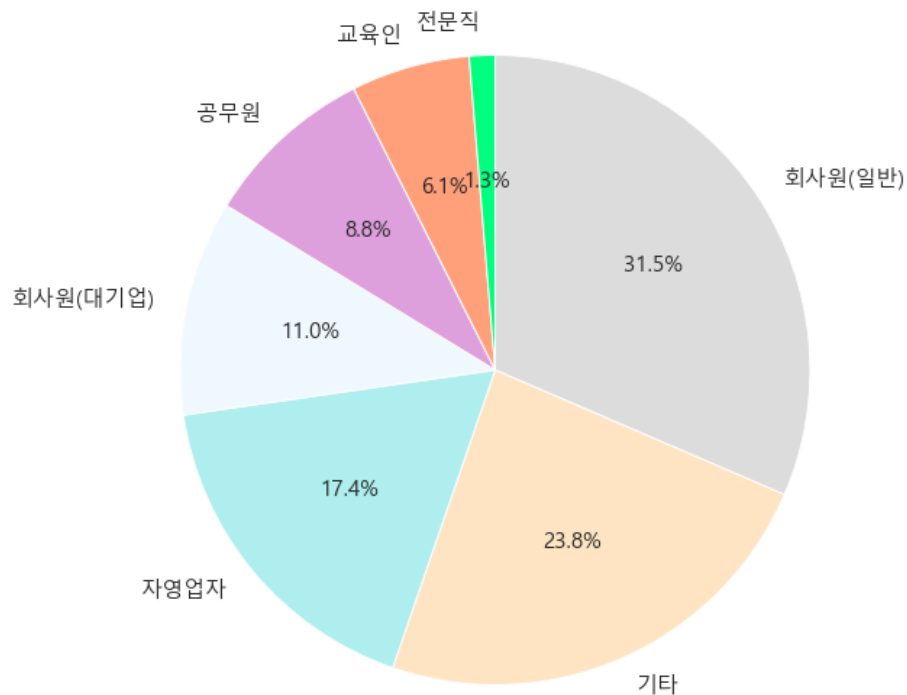
사전에 매우 잘 정제된 데이터

1. 데이터 상 결측치(NaN 등) 값 전무
2. 19.12~20.05와 20.12~21.05 기간의 데이터 비율도 거의 1:1 비율
3. 각 행은 동일 조건(거주지, 직업, 연령/성별 등)으로 **준식별화** 된 고객들 모음
4. 따라서 CNT column 값이 지나치게 높다 해서 무조건 데이터 상 제외하면 안 됨.

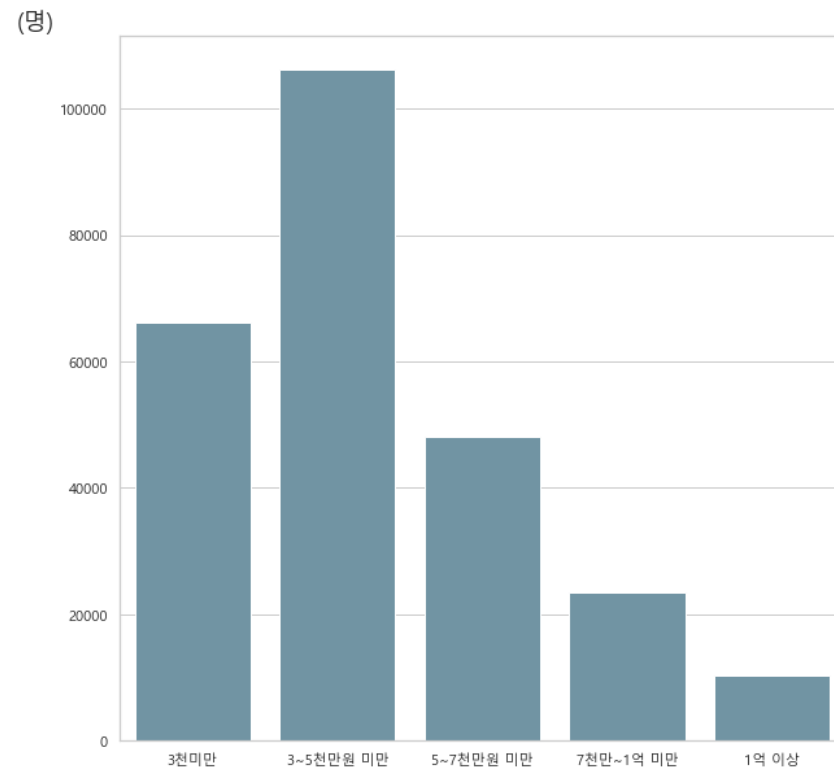
1. 데이터 특징



1인가구가 전체 데이터의
약 25%를 차지



회사원이 약 42% 차지
기타의 비율 또한(무직, 학생
등) 높은 것을 알 수 있음



3~5천만원 이상의 소득
가장 많은 분포(약 41%)

2. 업종 column 데이터 재분류화

```
def classification_df(data):

    data_b = data.copy()
    data_b['가전/주거관리'] = data_b[["CNT_가전", "CNT_공과금", "CNT_인테리어", "CNT_주생활관리"]].sum(axis=1)
    data_b['건강/의료'] = data_b[["CNT_건강보조", "CNT_동물병원", "CNT_병원"]].sum(axis=1)
    data_b['교육'] = data_b[["CNT_교육기타", "CNT_유아교육", "CNT_학교", "CNT_학원"]].sum(axis=1)
    data_b['교통'] = data_b[["CNT_교통", "CNT_대중교통"]].sum(axis=1)
    data_b['금융/보험'] = data_b[["CNT_보험"]].sum(axis=1)
    data_b['도소매업/쇼핑'] = data_b[["CNT_마트할인점", "CNT_면세점", "CNT_문구사무용품",
                                     "CNT_쇼핑기타", "CNT_편의점"]].sum(axis=1)

    data_b['뷰티/취미'] = data_b[["CNT_뷰티", "CNT_상품권", "CNT_서적", "CNT_취미", "CNT_화장품"]].sum(axis=1)
    data_b['숙박'] = data_b[["CNT_숙박"]].sum(axis=1)
    data_b['예술/스포츠/액티비티'] = data_b[["CNT_골프장", "CNT_관람", "CNT_놀이공원",
                                              "CNT_스포츠시설", "CNT_스포츠의류용품", "CNT_여행상품", "CNT_가례"]].sum(axis=1)
    data_b['온라인서비스'] = data_b[["CNT_배달앱", "CNT_온라인쇼핑", "CNT_정기결제"]].sum(axis=1)
    data_b['유흥/주류'] = data_b[["CNT_유흥", "CNT_주류판매점"]].sum(axis=1)
    data_b['요식업'] = data_b[["CNT_음식점", "CNT_제과점", "CNT_커피음료"]].sum(axis=1)
    data_b['자동차'] = data_b[["CNT_주유", "CNT_주차", "CNT_차량관리", "CNT_차량구매"]].sum(axis=1)

    data_b['패션/의류'] = data_b[["CNT_의류", "CNT_패션잡화", "CNT_백화점", "CNT_아울렛"]].sum(axis=1)

    data_b = data_b.drop(data.columns[27:], axis=1)
    data_b['소비년월'] = data_b['소비년월'].astype(str)

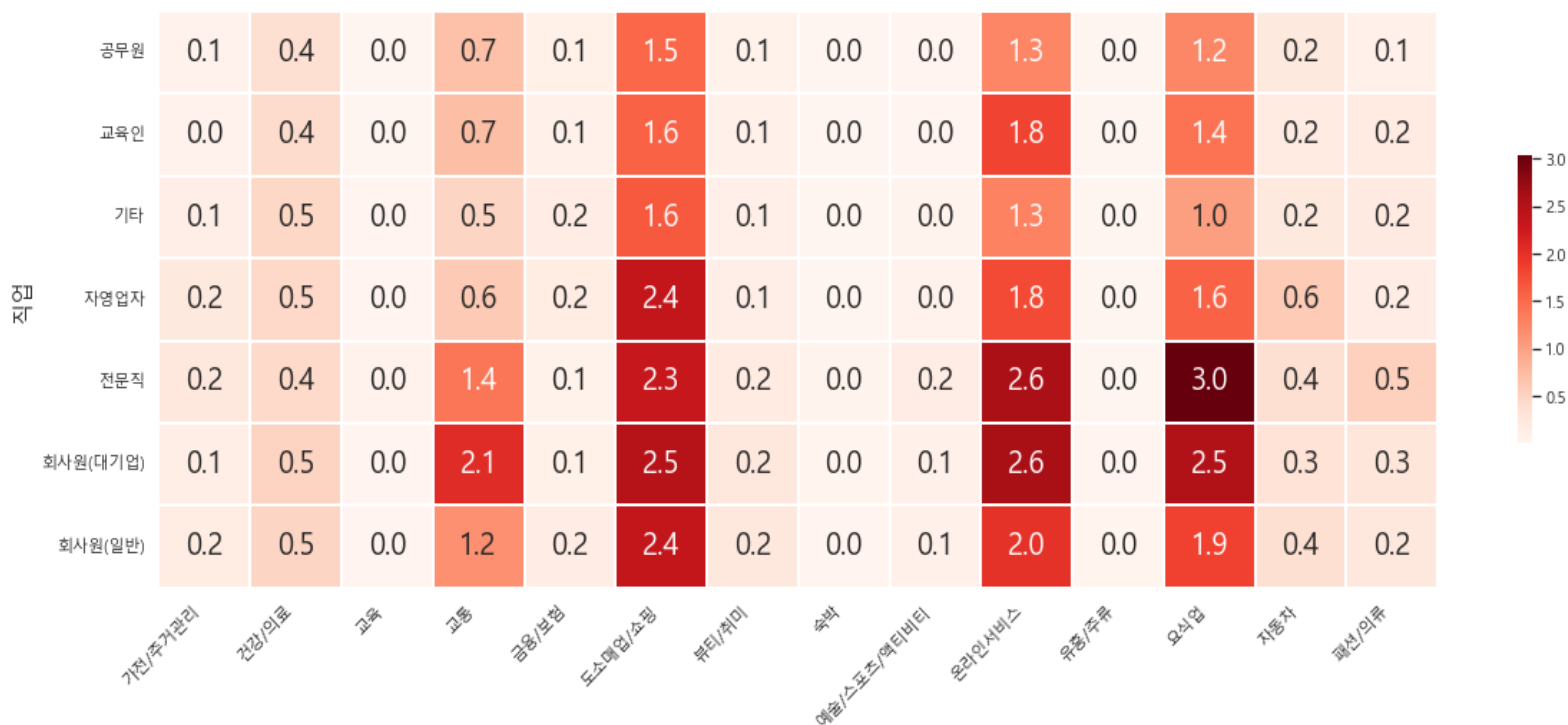
    return data_b
```

48개 => 14개로

- 업종 데이터의 종류가 많음(48개)
- 데이터를 탐색하다 보니 비슷한 성격의 업종이 많고, 매출 건수가 매우 적은 경우 (ex. CNT_가례)도 있어 재분류
- 다음 코드로 비슷한 업종끼리 묶어 48개의 업종을 각각의 업종을 잘 대표할 수 있는 14개로 바꿈
- Ex) CNT_교통, CNT_대중교통 => 교통

3. HeatMap 활용 데이터 상세 분석

1인당 직업별 매출 건수(초기)



매출 변수 다각도 관찰

- 1인당 업종별 매출 건수를 값 차이, 변화 등을 한 눈에 알아보기 쉽게 **Heatmap** 으로 나타냄
- 직업, 소득, 1인가구유무 등을 기준으로 두고 세부 분석
- 업종별로 매출 편차가 나타나는 경향이 있기 때문에 더 정확한 분석을 하기 위해 **1인당 매출 건수로 기준을 잡음**
- 공통적으로 인당 매출 건수가 높게 나타난 업종은 **요식업, 쇼핑/도소매, 온라인, 교통 관련 업종**

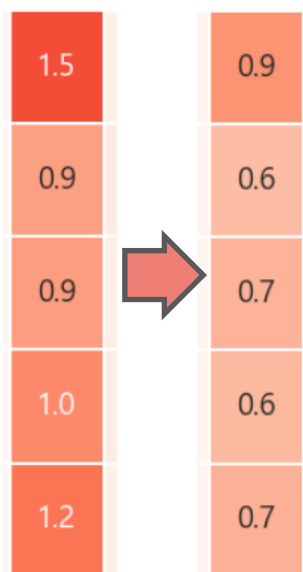
3. HeatMap 활용 데이터 상세 분석

```
def draw_business_heatmap(data, ylabel, title, size=(20,7)):  
    fig, ax = plt.subplots(figsize=size)  
    sns.heatmap(data,  
                cmap='Reds', cbar=True, cbar_kws={"shrink": .5},  
                linecolor="w", linewidths=1.5,  
                # 값 표기(소수 첫째자리까지)  
                annot=True, fmt='.1f', annot_kws = {'fontsize' : 20},  
                square=False)  
  
    # ticks 속성 상세설정  
    ax.set_xticklabels(ax.get_xticklabels(), rotation=45, horizontalalignment='right')  
    ax.set_yticklabels(ax.get_yticklabels(), rotation=0, horizontalalignment='right')  
    # label  
    plt.ylabel(ylabel, fontdict={'fontsize' : 15}, rotation=90, loc='center')  
    plt.title(title, fontdict={'fontsize' : 30}, pad=30)  
  
    return fig
```

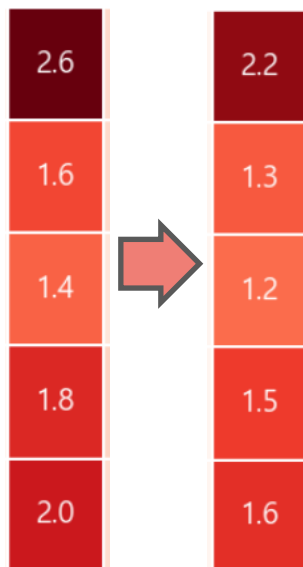
코드 해설

- 다양한 기준으로 분석할 것이기 때문에 함수화 시켜서 heatmap 객체 반환하는 방식 사용
- data와 기준(ylabel), 제목(title), heatmap 크기(size)를 매개변수화
- linewidth, linecolor 등 *heatmap()*의 매개변수, 각 axis의 tick을 설정할 수 있는 *xticklabels()*, *yticklabels()* 등을 통해 가독성을 한 층 높임

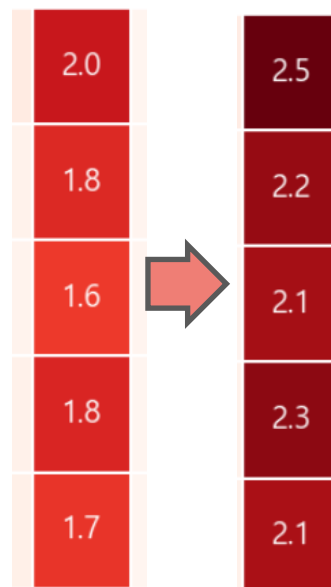
3. HeatMap 활용 데이터 상세 분석



교통



요식업



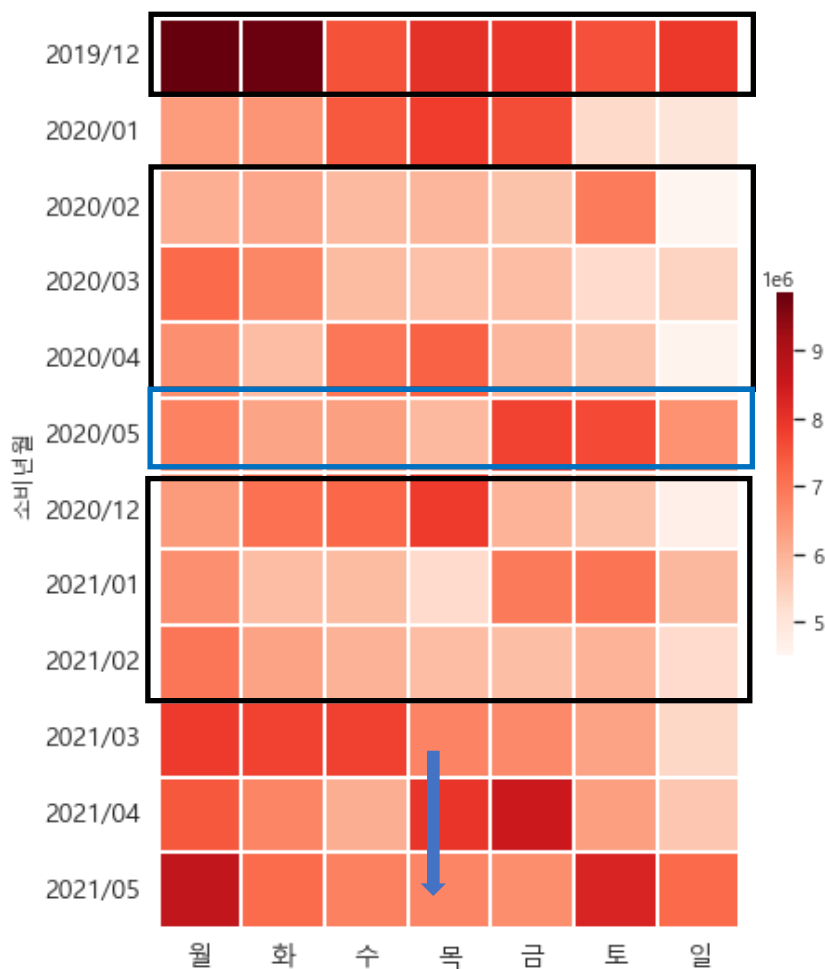
온라인서비스

기준 무관 공통적인 변화

- 현재 heatmap 자료는 소득 기준 예시
- 조건-업종별 분석을 해보면 공통적으로 나타난 점이 있음
- 왼쪽은 코로나 초기(19.12~20.05)고 오른쪽은 코로나 진행(20.12~21.05) 데이터.
- 1인당 교통 (1.1-) 0.7), 요식업 (1.88-) 1.56) 매출 건수 감소
- 반면 온라인서비스 (1.7 -) 2.2) 매출 건수는 눈에 띄게 증가

4. 월별-요일별 매출 건수 분석

월별-요일별 매출 건수 합계



코로나 국내 발병 이전

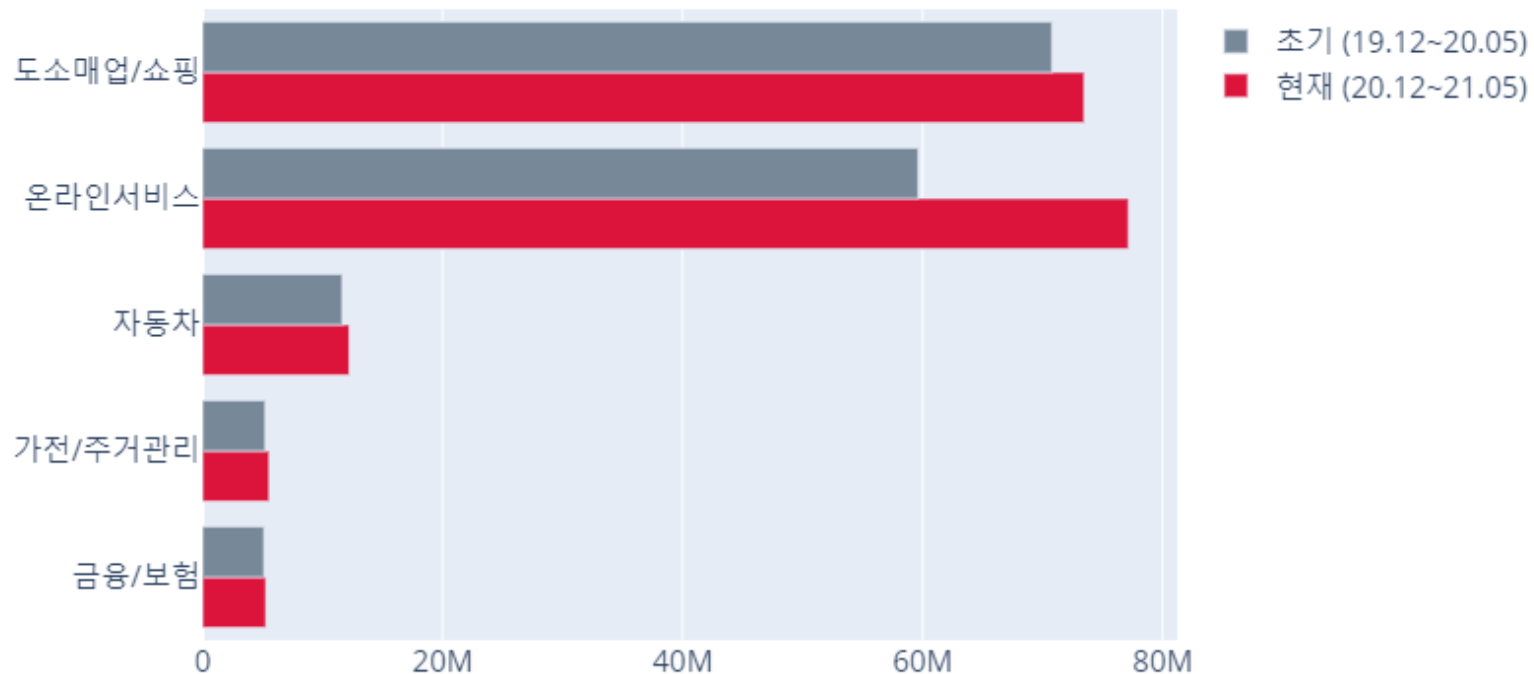
강력한 사회적 거리두기
(3월 일평균 214명)완화된 생활속 거리두기
(5월 일평균 23명)사회적 거리두기 2.5단계
(비수도권 2단계)
(12월 일평균 856명)백신 접종 시작
사회적 거리두기 2단계
(비수도권 1.5단계)

코로나19 상황을 중심으로

- 2020년 1월 국내 첫 코로나 확진자가 나오면서 사회적 거리두기 시작
- Heatmap과 거리두기의 관계를 보면
거리두기가 강화될 때, 매출 건수가 감소하는 양상을 띄고 있음
- 반대로 거리두기가 완화될 때는 매출 건수가 증가하는 모습
- 2021년 3월부터 부터 거리두기도 완화되며 백신 접종이 시작되면 '위드코로나' 움직임도 일어나게 됨 => 증가추세

5. 호황 업종별 매출 건수

호황 업종별 매출 건수 (단위: 백만)

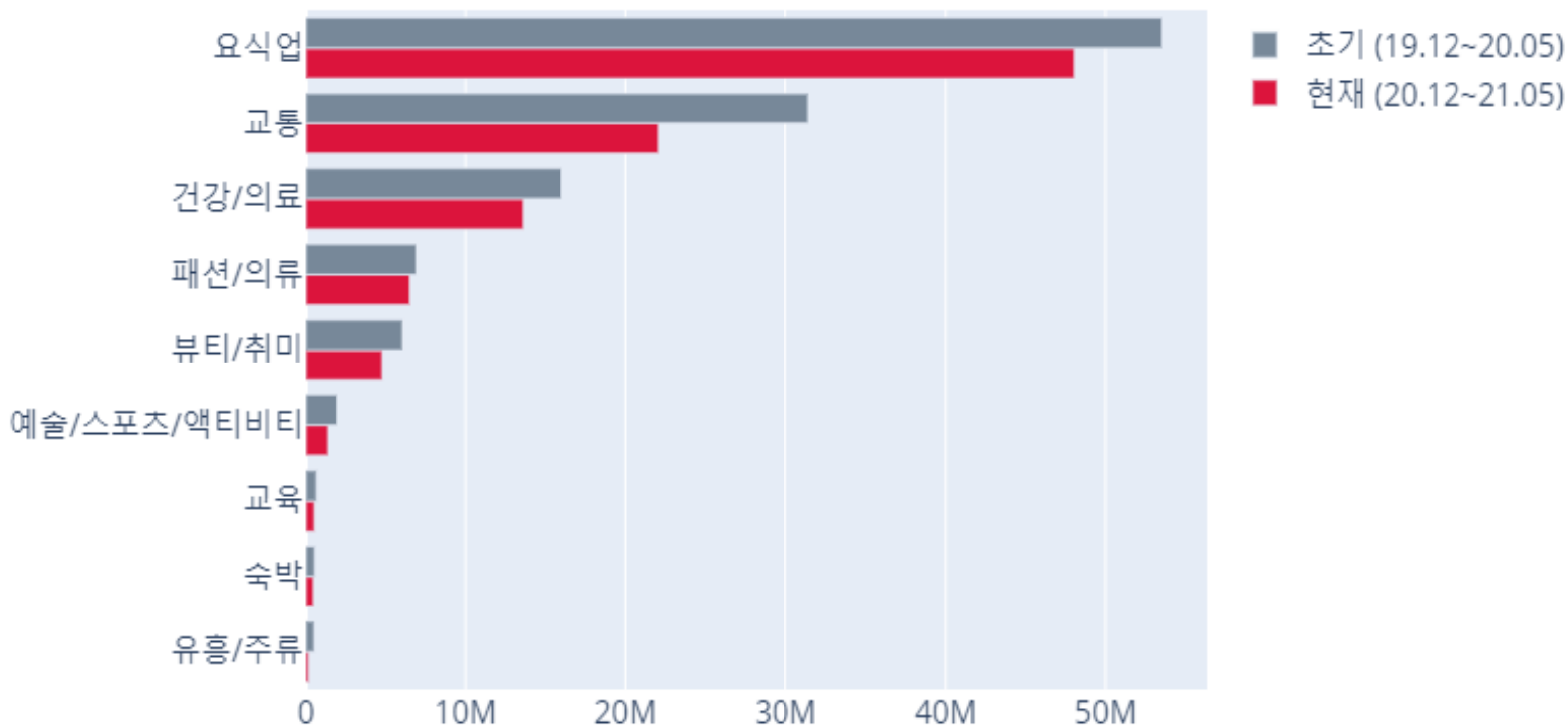


막대그래프 시각화

- 다음 그래프는 초기 대비 현재 총 매출 건수가 증가한 업종만 표시한 그래프
- 각 업종이 얼마나 타 업종 대비 상대적으로 매출 건수를 많이 차지하는지 비교, 초기 대비 증가한 현황을 보이게 하는 목적
- 도소매업, 온라인서비스, 자동차, 가전/주거관리, 금융/보험 업종이 초기 대비 매출건수 증가
- 실제로 코로나로 인해 인테리어 수요 증가(주거관리), 온라인서비스 증가, 차량 구매 증가 등을 기사로 본 적 있었는데 데이터 분석 결과와 매우 유사함

5. 불황 업종별 매출 건수

불황 업종별 매출 건수(단위: 백만)

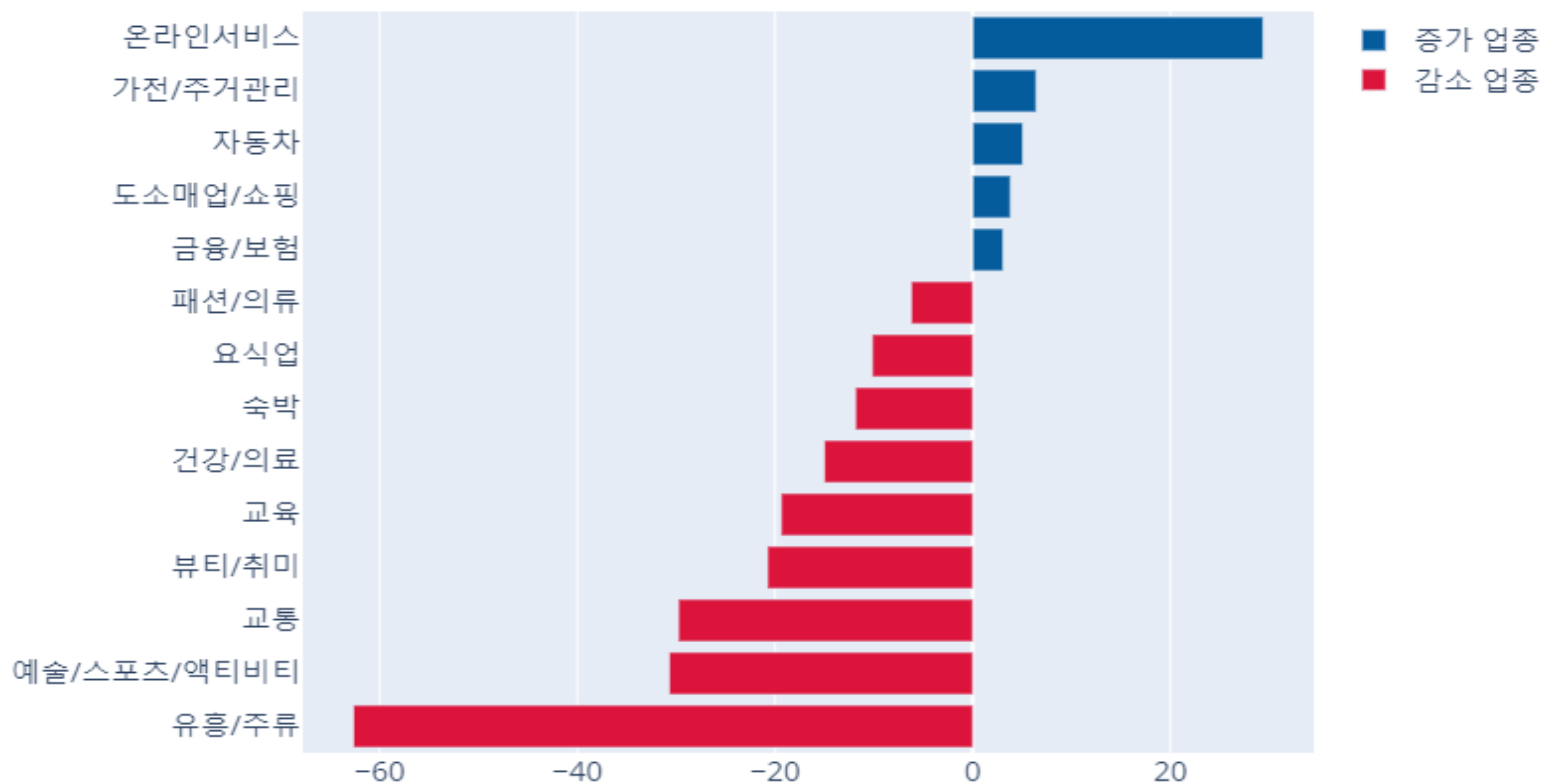


막대그래프 시각화

- 다음 그래프는 초기 대비 현재 총 매출 건수가 감소한 업종만 표시한 그래프
- 요식업, 교통, 예술/스포츠/액티비티 등 업종의 매출 건수가 감소함
- 코로나로 인해 음식점 등 자영업, 숙박, 교통관련 산업이 침체를 겪고 있다는 기사 또한 접했는데 데이터 시각화 결과에서도 그대로 나타남

5. 업종별 매출 건수 증감률

업종별 매출건수 변화율 (단위: %)



막대그래프 시각화(증감률)

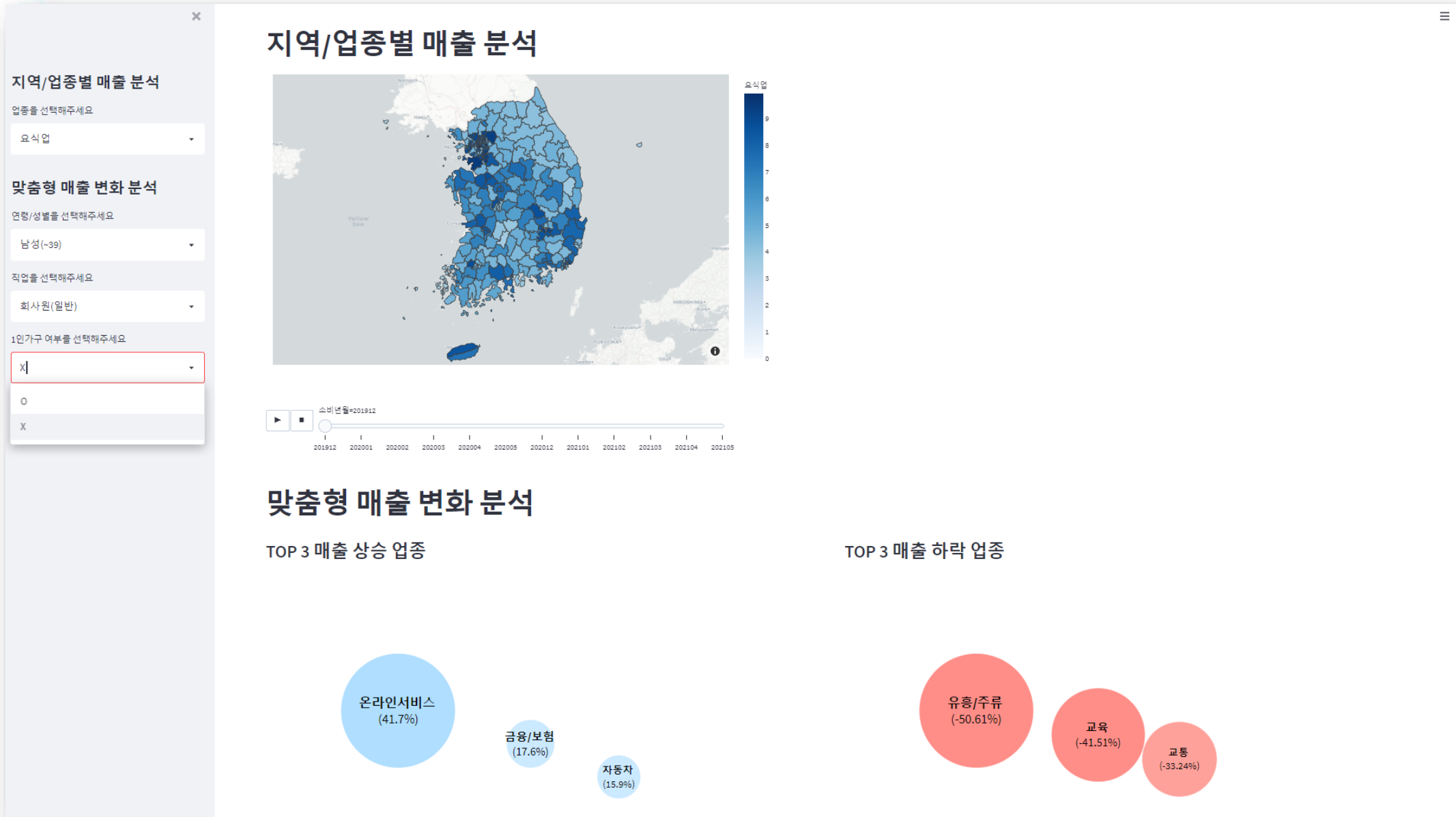
- 단순 수량 비교 그래프만으로 얼마나 증감했는지 시각적으로 파악하기 어려움
- 따라서 다음과 같이 코로나 초기 대비 현재 데이터의 증감률을 그래프로 시각화
- 온라인서비스 업종 (+29%) 의 매출건수가 가장 큰 증가세, 유흥/주류 (-62%) 및 예술/스포츠/액티비티 (-30%) 등 업종에서 큰 감소세를 보임

5. 업종별 매출 건수 그래프

```
def draw_bar(x_data, y_data, title, size):  
  
    fig = go.Figure()  
    fig.add_trace(go.Bar(y=y_data, x=x_data['현재'],  
                        orientation='h',  
                        marker_color='crimson',  
  
                        name='현재 (20.12~21.05)'))  
  
    fig.add_trace(go.Bar(y=y_data, x=x_data['초기'],  
                        orientation='h',  
                        marker_color='lightslategrey',  
  
                        name='초기 (19.12~20.05)'))  
  
    # 추가적인 layout 설정  
    w, h = size  
    fig.update_layout(  
        title_text=title, title_x=0.5,  
        autosize=False,  
        width=w,  
        height=h,  
        legend={'traceorder': 'reversed'},  
        font=dict(size=15)  
    )  
  
    return fig
```

코드 해설

- 마우스를 통해 실시간으로 형태가 변하는 그래프(Interactive Graph) 구현 위해 plotly 라이브러리를 사용
- `go.figure()`의 `add_trace()`를 통해 그래프를 추가하는 과정
- `add_trace()`의 파라미터 `orientation`을 통해 막대를 가로로 놓을 수 있음
- `fig.update()`를 통해 세부적인 레이아웃을 디자인



지역/업종별 매출 분석

업종을 선택해주세요

교육

맞춤형 매출 변화 분석

연령/성별을 선택해주세요

남성(~39)

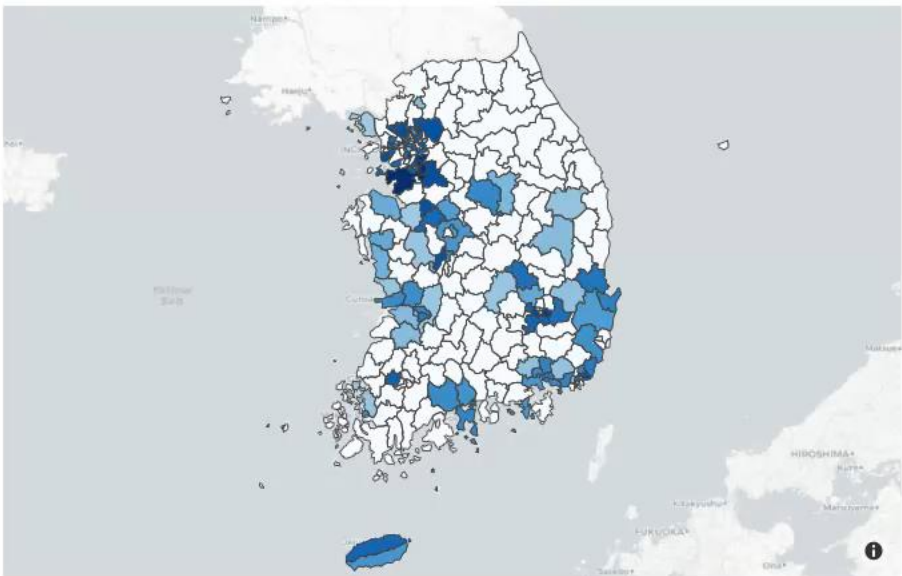
직업을 선택해주세요

회사원(일반)

1인가구 여부를 선택해주세요

0

지역/업종별 매출 분석



교육

6

5

4

3

2

1

0



맞춤형 매출 변화 분석

TOP 3 매출 상승 업종

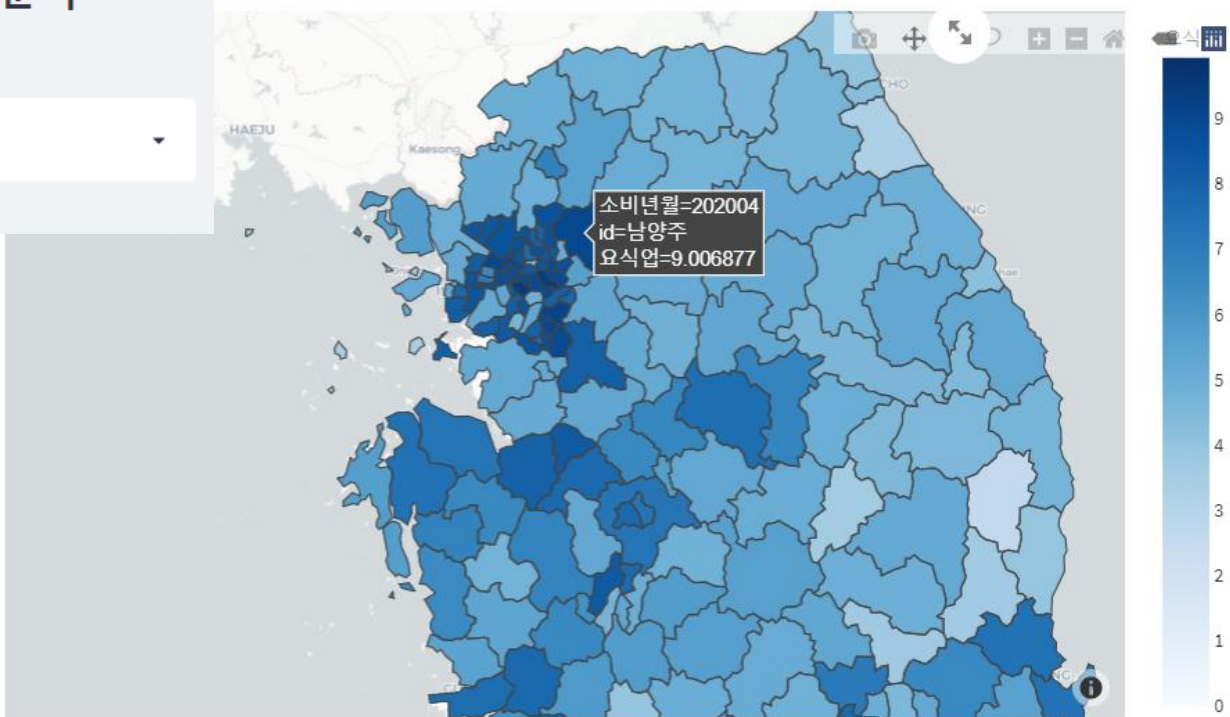
TOP 3 매출 하락 업종

1. 대시보드 기능 - 지역/업종별 매출 분석

지역/업종별 매출 분석

업종을 선택해주세요

요식업



주요 기능

- Streamlit 기반 프로토타입 제작
- Sidebar에서 조사하고 싶은 업종을 선택하여 시간대별, 지역별 변화 수치를 한 눈에 알 수 있음
- 더 명료한 변화를 시각화하기 위해 각 지역의 선택한 업종 매출건수 합계에 log를 취함
- 대한민국 지리 좌표 값이 담긴 json파일 토대로 구축

1. 대시보드 기능 - 맞춤형 매출 변화 분석

TOP 3 매출 상승 업종

맞춤형 매출 변화 분석

연령/성별을 선택해주세요

남성(~39)

직업을 선택해주세요

회사원(일반)

1인가구 여부를 선택해주세요

이

O

X

TOP 3 매출 하락 업종

온라인서비스
(40.4%)

상승: 40.4% trace 0

금융/보험
(18.8%)자동차
(7.1%)유흥/주류
(-56.29%)예술/스포츠/액티비티
(-46.89%)교통
(-35.14%)

주요 기능

- 연령/성별, 직업, 1인가구여부를 sidebar에서 선택하여 특성을 고려한 맞춤형 정보를 획득 가능
- 각 수치는 코로나 초기의 데이터와 현재의 데이터의 매출 건수의 증감률(%)
- 특성이 같은 집합 하의 고객들이 어떤 업종을 코로나 초기보다 많이 이용했는지, 줄었는지 시각적으로 쉽게 파악 가능

2. 대시보드 코드 - 1

```
with row_1:

    # 데이터 업로드
    data_b = pd.read_csv('./data/data_for_geo.csv')
    with open('./data/geo_korea.json', encoding='utf8') as f:
        geo_korea = json.load(f)

    business = sorted(data_b.columns[9:-1].unique())

    st.sidebar.title('지역/업종별 매출 분석')
    option_business = st.sidebar.selectbox('업종을 선택해주세요', tuple(business))

    st.title("지역/업종별 매출 분석")
    fig_map = px.choropleth_mapbox(data_b, geojson=geo_korea, locations='id', color=option_business,
                                   color_continuous_scale="Blues",
                                   range_color = (0, max(data_b[option_business])),
                                   mapbox_style="carto-positron",
                                   zoom=5.5, center={"lat": 36.0402, "lon": 127.4899},
                                   opacity=1,
                                   labels={'CNT': 'CNT'},
                                   animation_frame= '소비년월',
                                   animation_group = 'id',
                                   width=800,
                                   height=600,
                                   )

    fig_map.update_geos(fitbounds="locations", visible=False)
    fig_map.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
    st.plotly_chart(fig_map)
```

코드 해설(지역/업종 기능)

- 우선 지도를 그리기 위해 사전에 재가공한 data_b와 대한민국 좌표가 담긴 geo_korea.json 파일을 load
- Plotly 라이브러리의 choropleth_mapbox()를 이용해 Selectbox에서 값이 들어오면 그 값을 인자로 받아 반응형 map 그래프를 그림
- range_color 인자값 설정을 다음과 같이 한 이유는 업종의 전체 데이터의 최대값으로 설정해 타임라인에 따른 매출량 변화 파악 용이

2. 대시보드 코드 - 2

```
def custom_dashboard(data_b, age='B', job='A.전문직', sole='0'):
    garbage = data_b.columns[8:9]

    data_selected = data_b[(data_b['연령성별'] == age) & (data_b['직업'] == job) & (data_b['일인가구여부'] == sole)]
    data_selected = data_selected.drop(garbage, axis=1)

    data_selected_melt = data_selected.melt(id_vars=list(data_selected.columns[:8]) + ['id'],
        var_name="업종",
        value_name="이용횟수")

    data_selected_before = data_selected_melt[data_selected_melt['소비년월'] < 202011]
    data_selected_after = data_selected_melt[data_selected_melt['소비년월'] > 202011]

    groupby_before = data_selected_before.groupby(['업종']).sum()['이용횟수']
    groupby_after = data_selected_after.groupby(['업종']).sum()['이용횟수']

    # 업종별 ranking
    ranking = ((groupby_after - groupby_before) / (groupby_before) * 100).sort_values()
    ranking.dropna(inplace=True)

    top_3 = ranking[-3:].sort_values(ascending=False)
    down_3 = ranking[:3]

    # st.write(ranking)

    top_3_name, top_3_value = value_make(top_3, top=True)
    down_3_name, down_3_value = value_make(down_3, top=False)

    return top_3_name, top_3_value, down_3_name, down_3_value
```

코드 해설(맞춤형 기능)

- 맞춤형 기능은 증가율 top_3, 감소율 top_3를 구하는 *custom_dashboard()* 와 시각화하는 *draw_dashboard()*로 이루어짐
- Pandas의 *melt()*를 활용하여 column 값들을 분석에 용이하도록 재구조화
- 이 코드에서는 입력값으로 들어온 값들을 조건으로 하여 순위를 구하여 값과 업종명을 출력하는 코드

2. 대시보드 코드 - 3

```
def dashboard_draw(top_3_name, top_3_value, top):
    if top == True:
        hover_text = '상승'
    else:
        hover_text = '하락'

    x_coors = [0.5, 2, 3]
    y_coors = [3, 2.9, 2.8]

    color = color_selector(top_3_value, top=top)
    size=[val**2 for val in top_3_value]
    # layout 그리기
    layout = go.Layout(
        plot_bgcolor="#FFF", # Sets background color to white
        xaxis=dict(
            linecolor="#BCCDCD", # Sets color of X-axis line
            showgrid=False, # Removes X-axis grid lines
            visible=False, # numbers below
        ),
        yaxis=dict(
            linecolor="#BCCDCD", # Sets color of Y-axis line
            showgrid=False, # Removes Y-axis grid lines
            visible=False, # numbers below
        ),
        hoverlabel=dict(
            font_size=16,
            font_family="Rockwell"
        )
    )

    fig = go.Figure(
        layout=layout,
        data=[go.Scatter(

            # text 및 원 위치할 좌표
            x = [x_coors[idx] for idx in range(len(top_3_value))],
            y = [y_coors[idx] for idx in range(len(top_3_value))],
            text = [round(val, 1) for val in top_3_value],
            # x=[0.5, 2, 3], y=[3, 2.9, 2.8], text=[round(val, 1) for val in top_3_value],
```

코드 해설(맞춤형 기능)

- `draw_dashboard()`에서는 증가율 `top_3`와 감소율 `top_3`를 인자로 받아 시각화
- plotly 라이브러리의 graph를 담당하는 `go` 모듈로 원을 시각화함
- `Top_3_value`에서 항상 3개의 값이 들어오라는 보장이 없음(경우에 따라서는 1,2개 들어올 때도 있음).
- 따라서 좌표 `x,y` 좌표를 입력값의 변수에 따라 유동적으로 조정

2. 대시보드 코드 - 4

```
def color_selector(values, top):
    colors_blues = ["#70c4ff", "#8cd0ff", "#a1d8ff", "#b5e0ff", "#bfe4ff", "#cce9ff" ]
    colors_reds = ["#ff4136", "#ff5d54", "#ff7870", "#ff958f", "#ffada8", "#ffc5c2"]
    colors = []

    if top == True:
        colors_original = colors_blues
    else:
        colors_original = colors_reds

    for val in values:
        tmp = abs(val)
        if tmp >= 60:
            colors.append(colors_original[0])
        elif tmp >= 35 and tmp <60:
            colors.append(colors_original[1])

        elif tmp >= 20 and tmp <35:
            colors.append(colors_original[2])

        elif tmp >= 10 and tmp <20:
            colors.append(colors_original[3])

        elif tmp >= 5 and tmp <10:
            colors.append(colors_original[4])

        else:
            colors.append(colors_original[5])

    return colors
```

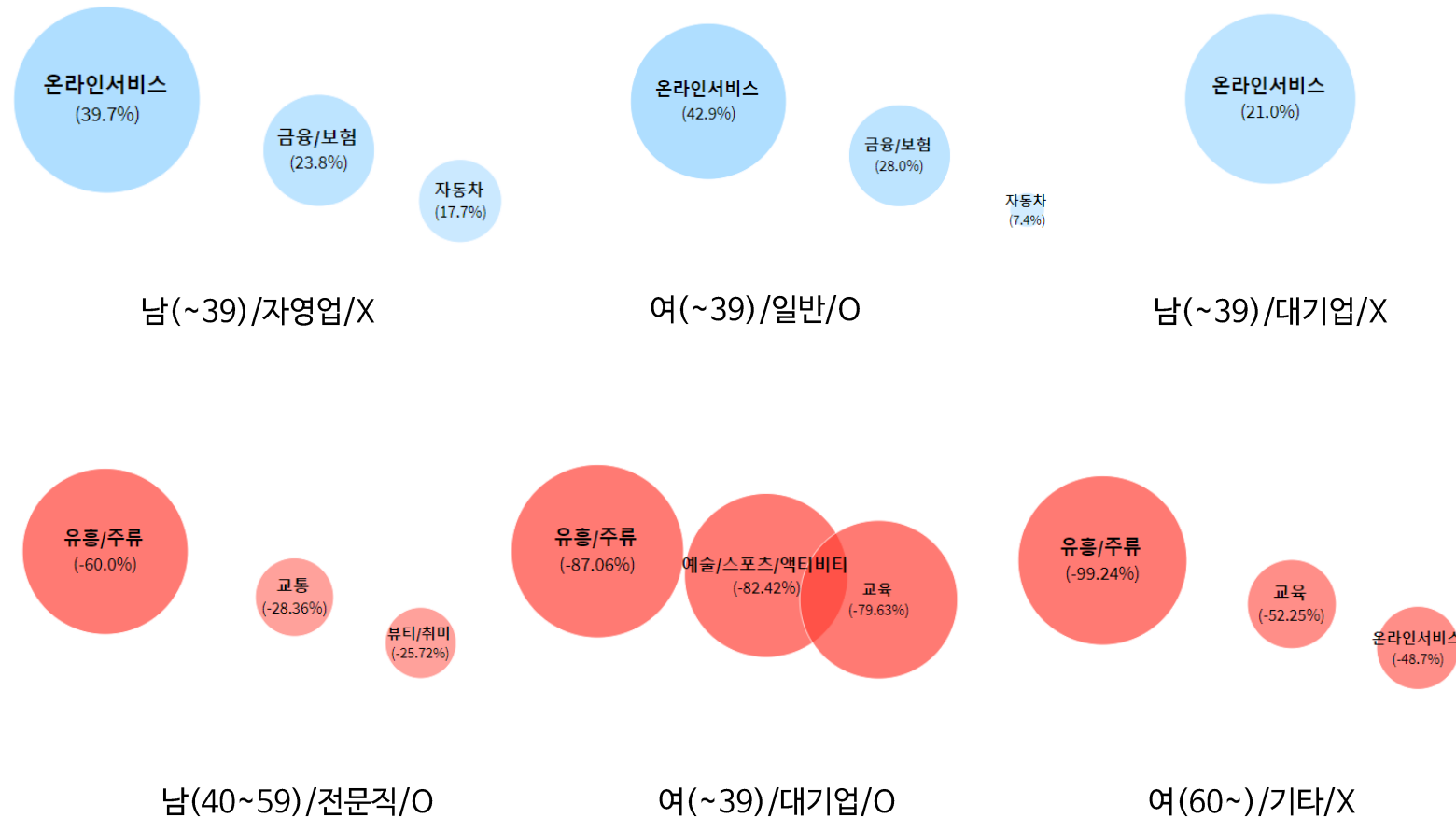
코드 해설(맞춤형 기능)

- `Color_selector()` 메소드는 증가율을 시각화 할 경우는 푸른색 계통, 감소율을 시각화 할 경우에는 붉은색 계통을 출력하기 위해 직접 정의함
- Top 파라미터를 통해 구분하며, 증감률 값에 따라서 색깔이 6단계로 구분되도록 설정함
- 반복문과 조건문을 통해 colors 리스트에 색깔이 담기고, 모든 색깔이 담겼으면 반환됨

4. 추가분석 / 프로젝트 결과



1. 대시보드를 통한 분석



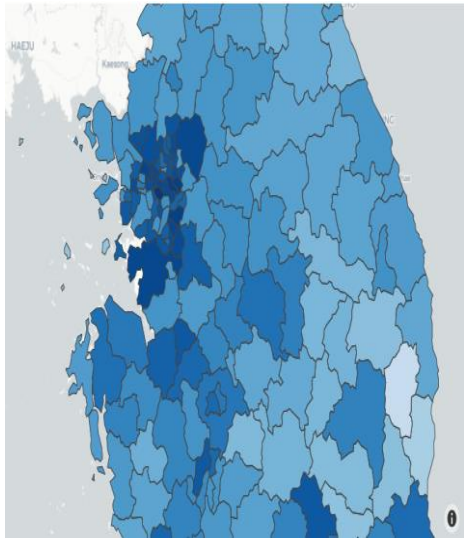
맞춤형 대시보드 분석

- 젊은 연령대(~39세)의 매출 상승 업종은 온라인서비스 업종의 증가세 및 빈도가 가장 높음

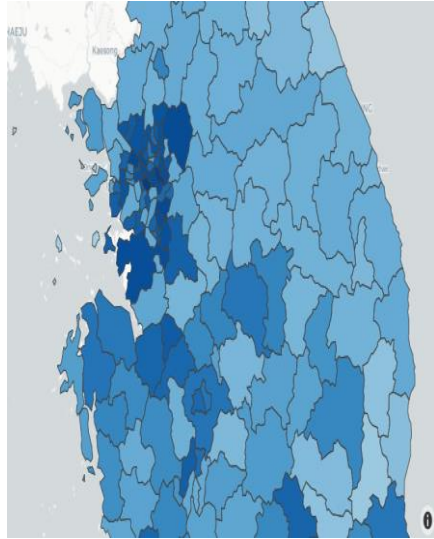
- 또한 성별과 연령, 직장 및 1인가구를 막론하고 유호/주류 업종의 감소세가 가장 크고 많았으며 교육 업종도 빈도가 큼

1. 대시보드를 통한 분석

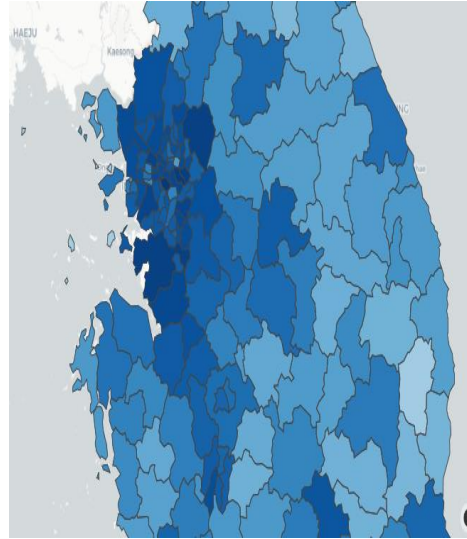
option = “요식업”



2020.01



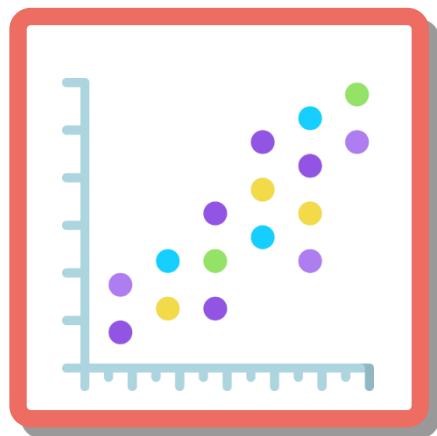
2020.03



2020.05

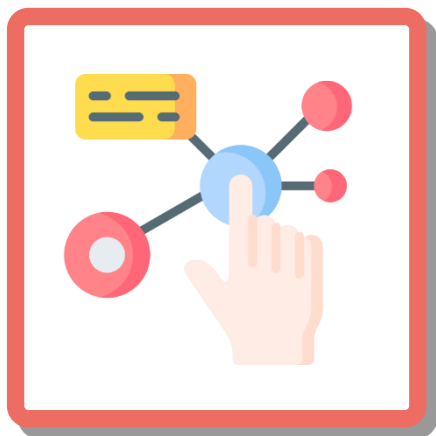
지리 대시보드 분석

- 2020년 1월부터 5월까지 요식업 업종을 기준으로 타임라인을 보여줌
 - 1월 : 국내 확진자 최초 발생
 - 3월 : 강화된 사회적 거리두기
 - 5월 : 완화된 사회적 거리두기
- 거리두기의 강화 등의 정책 변경에 따라 매출 건수(색의 농도) 또한 달라진 것을 관찰 가능
- 거리두기와 매출 증감 사이 모종의 관계가 있다고 추측할 수 있다



데이터 시각화 기법 공부

시각화 과정에서 효과적인 색 사용해
올바른 차트 그리는 방법을 습득
배운 내용을 적용하는 과정에서
꾸미는 방법을 습득



Interactive 시각화

마우스 등으로 조절이 되는 반응형
시각화를 해보고 싶었는데, 새로운
라이브러리(plotly)로 목표 이룸



대시보드 설계, 구축

실제로 대시보드를 구축해보고
사용해 보니 다양한 분석을 시도
해 볼 수 있었음.
지역/업종별 대시보드는 다른 방식의
시도를 통해 추가, 보완 필요



데이터와 현실 세계 접목

코로나 거리두기 정책 및 평균
확진자 등을 데이터에 접목해보니
다양한 방식으로 접근할 수 있었음
분석력을 한 층 키울 수 있었던 기회



보완점 & 목표

1. Tableau를 추가적으로 학습하여 대시보드 UI, 기능 개선
2. 코로나 확진자 데이터를 통해 업종별 매출 추가적인 분석 필요
3. Streamlit 기반 프로토타입을 실제 웹개발을 통해 배포
4. 다양한 통계적 기법을 시각화 기법과 접목

참고자료

- 「코로나 팬데믹이 한국경제와 산업에 미친 영향」 강두용 외 (2021),
- 코로나 이후 네 가지 주요변화에 주목해야(출처 : 2021.01.22.최종화.대한민국 정책브리핑(www.korea.kr))

외부 공공데이터

- geo_korea.JSON (출처 : 대한민국 최신 행정구역 (GIS DEVELOPER))

감사합니다

