

<산학캡스톤디자인1_중간보고서>

CAP: Corporates Analysis & Prediction of the risk

(중소기업벤처부 제공 데이터를 활용한 광주·전남소재 중소기업 도산 가능성 분석과 예측)

산학캡스톤디자인1_03분반

팀명 : CAPE(CAP-Engineers)

팀 구성원(총 4명)

- 임형열(IT융합대학 컴퓨터공학과, 20134888)
- 김동진(IT융합대학 컴퓨터공학과, 20144701)
- 조재혁(IT융합대학 컴퓨터공학과, 20144807)
- 김태완(IT융합대학 컴퓨터공학과, 20144817)

담당교수 정현숙(IT융합대학 컴퓨터공학과)

목 차

1. 서 론

- 1-1. 개발동기 -----(1)
- 1-2. 개발형태 -----(1)
- 1-3. 기존 관련연구 및 차이점 -----(1)
- 1-4. 기대효과 -----(1)

2. 본 론

- 2-1. 팀원 소개 및 역할 -----(2)
- 2-2. 데이터 소개 및 선정 이유 -----(2)
- 2-3. 개발환경 -----(2)
- 2-4. 기본설계 -----(3)
- 2-4. 구현방식 -----(6)

3. 결 론

- 3-1. 일정표 및 주차별 세부 계획 -----(6)
- 3-2. 구현방법(세부) -----(7)
- 3-3. 협업방식 및 팀원 소감 -----(12)

4. 부 록

- 4-1. 참고 서적과 관련 자료 -----(13)

1. 개발목적

1-1. 개발동기

최근 코로나 19로 인한 전 세계 경기침체 현상으로 유동성 위기와 같은 자금 흐름의 압박을 견디지 못하고 도산하는 기업¹⁾이 늘어나는 추세임. 또 특정 기업이 자신의 위험 요소에 대해 자세히 파악하지 못하고 있는 경우가 많고, 투자자 입장에서 해당 기업에 투자할 경우 위험성에 대한 자료를 제공받기 힘들. (대기업이나 중견기업의 경우 평가자료를 손쉽게 얻을 수 있지만 중소기업은 상대적으로 자료를 수집하기 어려움.) 특히나 기존의 시장은 상장기업을 중심으로 분석 자료를 제공하는 경우가 많으므로 대부분 비상장기업이 많은 중소기업은 분석 자료가 존재하지 않거나 구하기 힘들.

이에 본 팀은 각 기업의 재무상황과 위험요소 등을 평가한 데이터를 활용하여 기업의 도산 가능성을 분석해보고 예측해보고자 함. (데이터 출처와 세부 형태는 “2-2. 데이터 소개 및 선정 이유” 참고)
본 분석을 통해 기업 차원에서 자신의 취약점(위험요소)을 직관적으로 파악, 대응책을 마련할 수 있으며 개인 투자자도 이를 통해 투자하고자 하는 기업에 대한 평가자료를 쉽게 접하고 투자에 이용할 수 있음.

프로젝트 이름 CAP는 Corporates Analysis & Prediction of risk의 약자로서 데이터셋을 통한 “중소기업의 위험성 분석 및 예측”이라는 주제를 줄인 뜻임.

1-2. 개발형태

본 프로젝트는 웹 서비스 형태로 구현하는 것이 목표임. (개발 환경 및 세부 형태, 설계는 각각 “2-3. 개발환경”, “2-4. 기본설계”, “2-5. 구현방식” 부분을 참고)

1-3. 기존 관련연구 및 차이점

규모가 큰 포털사이트(네이버, 다음 등)에서 제공하는 주식 관련 페이지에서 제공하는 주가 관련 자료가 있음. 그러나 본 자료의 경우 실시간으로 변화하는 특정 기업의 주가를 보여줄 뿐 해당 기업에 영향을 줄 수 있는 요소 분석이나 예측 부분은 들어가 있지 않음. 또한, 시중에 존재하는 기업 컨설팅 자료의 경우 규모있는 기업(대기업, 중소기업)이거나 시장에 상장된 기업을 대상으로 한 자료만 존재함. 중소기업은 대부분 비상장기업이므로 자료가 없거나 있더라도 부실한 수준임. 본 팀은 이러한 점에 주목하여 중소기업을 대상으로 한 분석 자료와 도산 가능성을 예측해보고자 함.

1-4. 기대효과

해당 분석을 통해 특정 기업은 취약점(위험 요소)을 가시화된 자료로 쉽게 파악하고 대응책을 마련할 수 있음. 개인 투자자는 해당 자료를 활용하여 건전한 재무상태를 가진 기업을 쉽게 선별할 수 있으므로 투자 위험도를 줄일 수 있음. 특히 중소기업 관련 분석자료(통상적으로 중소기업은 비상장기업이 많음)는 인터넷에서 쉽게 구하기 힘들. 본 프로젝트를 통해 투자자가 중소기업에 대한 여러 자료를 좀 더 쉽고 편리하게 접근할 수 있음. 더해서, 본 자료를 활용하여 국가 차원에서도 기업에 대한 맞춤형 정책을 수립할 수 있다는 장점이 있음.

1) 본 프로젝트에서의 기업은 대기업/중견기업이 아닌 중소기업을 지칭함.

2. 개발준비

2-1. 팀원 소개 및 역할

CAPE²⁾ 팀은 총 4명으로 구성되어 있으며, 세부 사항은 다음과 같음.

임형열(IT융합대학 컴퓨터공학과(13), 팀장, doodleima@naver.com)

- 자료 수집, Python을 활용한 위험도 예측

김동진(IT융합대학 컴퓨터공학과(14), engadoridori@gmail.com)

- Python을 활용한 위험도 분석, 코드 자동화

조재혁(IT융합대학 컴퓨터공학과(14), jihst2285@naver.com)

- Python을 활용한 코드 자동화, Django 기반 웹 서비스 제공을 위한 설계 및 구현

김태완(IT융합대학 컴퓨터공학과(14), ktwan0782@gmail.com)

- Kakao Oven을 활용한 기본 형태 설계, Django 기반 웹 서비스 제공을 위한 설계 및 구현

2-2. 데이터 소개 및 선정 이유

본 팀은 중소벤처기업부와 중소기업중앙회에서 주관하는 공모전에서 제공받은 중소기업 관련 평가요소 데이터셋을 기반 데이터로 사용함. 본 데이터는 중소벤처기업부 홈페이지의 통계자료 탭의 주제별 통계 >> 중소기업 관련 자료(모두)로도 제공하고 있는 상황임. 굳이 공모전에서 제공받은 데이터를 쓴 이유는 홈페이지에서 제공하는 자료를 모두 합친 형태였기 때문에 가공과정을 줄일 수 있어서였음. 그리고 본 자료는 국가기관에서 제공하는 데이터이므로 신뢰성 있는 데이터로 판단하였음.

(데이터 제공 중소벤처기업부, <https://www.mss.go.kr/site/smba/foffice/ex/statDB/temaList.do>)

2-3. 개발환경

본 프로젝트 구현에 사용한 개발환경은 다음과 같음.(이하 모든 인원 동일)

- 운영체제 : Microsoft Windows 10 Education(빌드 18362)
- 프로그래밍 언어 : Python 3.7 (<https://www.python.org/>)
- 웹 서비스 구현 도구 : Web Framework 'Django' (<https://www.djangoproject.com/>)
- IDE : 총 2개 사용
 1. Anaconda3: Jupyter Notebook(Python) (<https://jupyter.org/index.html>)
 2. Visual Studio Code 1.45(Django) (<https://code.visualstudio.com/>)

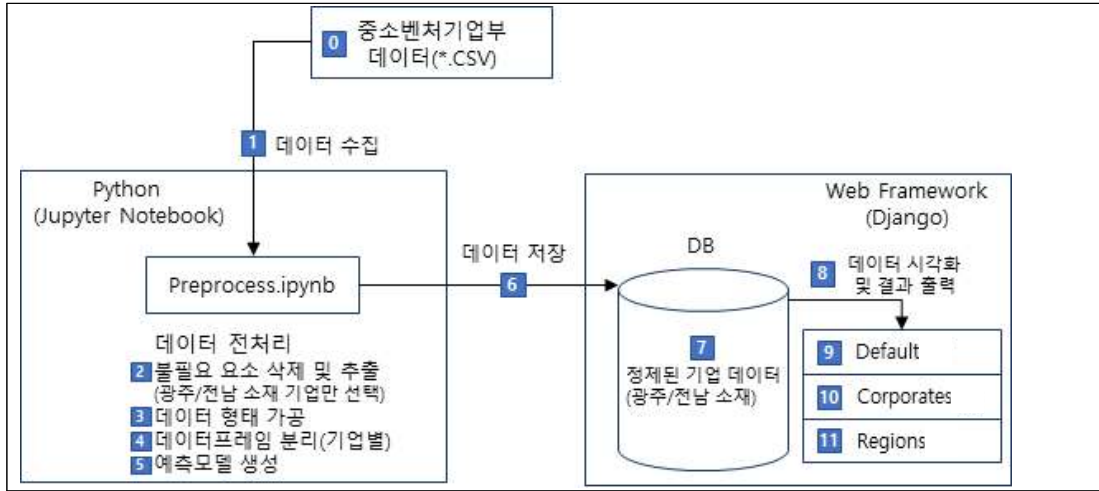
프로그래밍 언어 Python 3.7은 머신러닝에 강점을 지닌 언어이며, 인터프리터 방식을 활용하여 해당 코드에 대한 결과를 즉시 확인할 수 있다는 장점이 있음. 그리고 이를 기반으로 한 IDE(통합개발자도구) Anaconda3의 Jupyter Notebook은 Python의 장점을 극대화한 도구임. 코드의 실행결과가 코드와 같이 저장되어 있으며 하나의 화면에서 시각화 도구까지 활용할 수 있다는 장점이 있음.

또한 백엔드 환경으로 Python을 가진 웹 프레임워크 Django의 경우 모델 - 템플릿 - 뷰 구조로 되어 있음. 즉 개발자가 원하는 모양으로 데이터(모델)를 받고 템플릿 웹 사이트 구성요소를 그대로 사용할 수 있으며, 뷰로 Python을 내부 백엔드로 사용 가능한 장점을 가짐.

2) 프로젝트 이름 CAP에 Engineers의 약자인 E를 붙임. CAP를 만드는 Engineers라는 뜻.

2-4. 기본설계 (1/3)

다음은 구현하려고 하는 형태의 기본 설계도임. (번호별 상세 설명은 그림 아랫부분을 참고)



0. 기반 데이터 : 중소기업부 웹사이트의 통계자료 → 주제별 통계의 '중소기업' 관련 데이터모음 (<https://www.mss.go.kr/site/smba/foffice/ex/statDB/temaList.do>)

1. 데이터 수집 : 중소기업부 웹사이트에서 데이터를 *.CSV 파일로 가지고 옴

- 1-1. 해당 데이터는 기업의 이름(코드)을 나타내는 부분과 평가요소(45개)를 하나의 행으로 가짐
- 1-2. 이러한 행이 중소기업부의 조사대상이 되는 중소기업의 총 개수만큼 존재(약 2천여 행)
- 1-3. 1-2와 같은 형태의 데이터가 총 12개의 CSV 확장자 파일로 존재함('19.2. ~ '20.1)
- 1-4. 총 12개의 CSV 파일을 Python(Jupyter Notebook)으로 불러옴(데이터 전처리 준비)

2. 불필요 요소 삭제 및 추출

- 2-1. 필요한 요소만 추출 : 데이터셋 중 광주/전남 소재 기업에 해당하는 내용만 선택하여 추출
- 2-2. 분석에 사용할 데이터 개수의 최소 기준을 선정, 그 이상 보유한 기업의 데이터만 보존
(데이터 결측치가 존재하지 않는 부분을 카운트하여 선정한 기준값 이상을 보유하고 있으면 해당 기업의 데이터 보존, 기준값 아래일 경우 해당 기업의 데이터 전체를 삭제)
- 2-3. 불필요 요소 삭제 : 도산 가능성 분석을 위해 요소(특징값) 선정(10개), 나머지 요소는 삭제

3. 데이터 형태 가공

- 3-1. 결측치 문제 해결 : 데이터에 존재하는 결측치를 채움
(각 분기별로 데이터를 그룹화하여 평균값을 산출한 후 이를 해당 분기의 결측값에 채움)
- 3-2. 중소기업부에서 부여한 임의의 기업코드를 규칙성 있는 숫자코드로 변환
(기업 소재지역코드(24/36) + 000n의 형태로 변환)

4. 데이터프레임 분리(기업별)

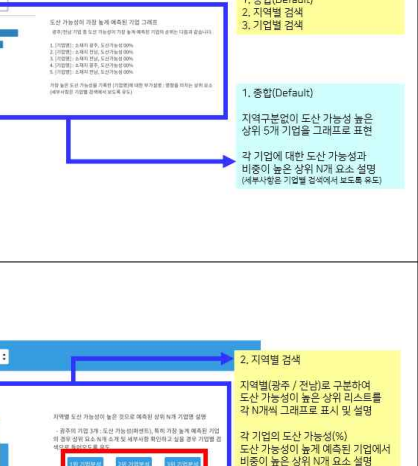
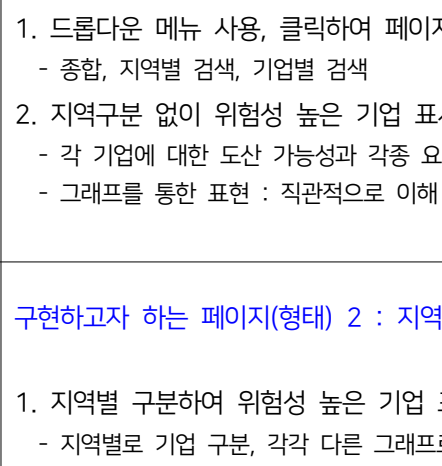
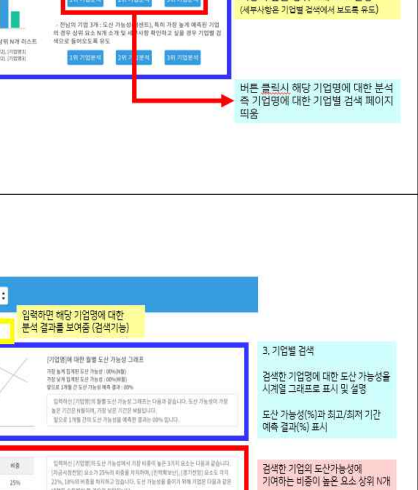
- 4-1. 같은 숫자코드를 갖는 데이터를 그룹화하여 해당 기업에 대한 데이터프레임 생성
같은 숫자코드는 같은 기업코드, 즉 같은 코드를 갖는 데이터는 같은 기업에 대한 데이터임
총 12개의 CSV 확장자 파일 중 같은 기업에 대한 데이터를 모을 경우 총 12개의 행이 존재하게 되며, 여기서 예측모델 검증을 위해 가장 마지막에 존재하는 행('20.1.에 해당)은 포함시키지 않음(테스트 데이터로 활용하기 위함)
공통 기업코드 값을 갖는 11개의 행을 하나의 데이터프레임으로 묶음

2-4. 기본설계 (2/3)

5. 예측모델 생성 : Facebook에서 개발한 fbprophet 라이브러리를 사용하여 예측값 생성
 - 5-1. 기업별 데이터프레임에 존재하는 요소에 가중치를 부여하여 도산 가능성 컬럼 생성(값 채움)
 - 5-2. fbprophet 라이브러리를 사용하여 미래 1개월('20.1)에 해당하는 예측모델을 생성
도산 가능성을 직접 예측하는 것이 아닌 데이터프레임의 10가지 요소를 예측모델로 생성
이후 4-1.에서 테스트 데이터로 활용하기 위해 따로 빼낸 데이터, 즉 실제로 가지고 있는
'20.1.에 해당하는 데이터와 예측된 요소를 대조하여 신뢰성을 판단함. 이후 기준 신뢰도에
부합할 경우 예측된 요소로 도산 가능성을 산출. 신뢰도에 부합하지 않을 경우 관련 요소와
설정을 변경한 후 기준 신뢰도에 부합하는 값이 나올 때까지 해당 과정을 반복
 - 5-3. 5-2.까지의 과정 완료시 기업에 대한 기존 11개 행 + 예측한 1개 행, 총 12개 행이 존재
 - 5-4. 5-3.까지의 과정을 모든 가공된 기업 데이터프레임에 반복하여 적용
 6. 데이터 저장 : 5번 과정까지 모두 완료된 데이터를 DB에 저장
 - 6-1. 웹 프레임워크 Django는 DB기능을 하는 도구³⁾를 내장하고 있으므로 별도 도구 사용
불필요
 7. 정제된 기업 데이터 : 광주/전남소재 기업 데이터(약 200개)
 - 7-1. 하나의 기업에 대한 데이터는 공통 기업코드를 갖는 11+1개월 어치의 행이 존재
 - 7-2. 각 행은 기업코드와 자문을 통해 선정한 10가지 요소(특징값)로 이루어져 있음
 8. 데이터 시각화 및 결과 출력 : 웹 프레임워크 사용
 - 8-1. 데이터를 시계열 그래프, 막대그래프로 시각화한 후 이 결과를 화면에 출력
 9. Default : 기본 홈 페이지(종합 페이지, 소재지역 상관없이 도산가능성이 높은 상위 기업과 설명)
 10. Corporates : 기업별 분석 페이지(특정 기업을 검색할 경우 해당 기업 분석과 설명 출력)
 11. Regions : 지역별 분석 페이지(광주/전남 각 지역의 도산가능성 높은 상위 기업과 설명)
- * 9~11에 해당하는 페이지 세부 설명은 아래의 웹 서비스 형태와 설명 부분 참고

3) SQLite - <https://sqlite.org>

구현하려고 하는 웹 서비스의 형태와 설명은 다음과 같음.⁴⁾

 <p>구현하고자 하는 페이지(형태) 1 : 종합(Home)</p> <ol style="list-style-type: none"> 드롭다운 메뉴 사용, 클릭하여 페이지 변경 <ul style="list-style-type: none"> 종합, 지역별 검색, 기업별 검색 지역구분 없이 위험성 높은 기업 표시(n개) <ul style="list-style-type: none"> 각 기업에 대한 도산 가능성과 각종 요소 설명 그래프를 통한 표현 : 직관적으로 이해 가능 	 <p>구현하고자 하는 페이지(형태) 2 : 지역별 검색</p> <ol style="list-style-type: none"> 지역별 구분하여 위험성 높은 기업 표시(n개) <ul style="list-style-type: none"> 지역별로 기업 구분, 각각 다른 그래프로 표시 각 기업별 위험성, 비중이 높은 요소 설명 세부사항은 기업별 검색에서 보도록 유도 버튼 클릭시 해당 기업에 대한 분석 페이지로 <ul style="list-style-type: none"> 해당 기업명에 대한 기업별 검색 페이지 띄움
 <p>구현하고자 하는 페이지(형태) 3 : 기업별 검색</p> <ol style="list-style-type: none"> 검색 : 입력시 해당 기업의 결과페이지 출력 검색한 기업에 대한 위험성을 설명 <ul style="list-style-type: none"> 시계열 그래프와 글로 설명 최고/최저 기록한 위험성과 예측한 값 표시 기여비중이 높은 상위 n개 요소 출력 <ul style="list-style-type: none"> 해당 요소와 그것들이 각각 차지하는 비중 위험성 감소를 위한 간단한 조언(의견) 표시 	

4) 카카오 오븐을 사용하여 웹 UI에 대한 기본적인 설계를 진행하였음.

2-5. 구현방식

1. '중소기업 통계데이터 활용정책 아이디어 공모전'에서 제공한 중소기업 API를 가지고 분석.
(해당 API는 지역별 기업 현황과 기업별 재무상태를 나타낼 수 있는 여러 지표를 보유하고 있음.
또한, 동일한 데이터를 중소벤처기업부 홈페이지에서도 오픈데이터로 제공 중)
2. 해당 API에서 광주/전남 지역 소재 기업들의 데이터만 선별.
3. 선별된 데이터의 기업의 각종 지표들 중 중요도가 높은 n개의 특징값을 추출.
4. 머신러닝 분류기를 사용하여 예측, 그래프를 이용하여 예측결과 시각화.
5. 웹 프레임워크를 활용, 분석 및 예측 결과자료를 웹 사이트에서 볼 수 있도록 함.
(세부 형태(설계도)와 방법은 각각 "2-4. 기본설계"의 기본 설계도와 "3-2. 구현방법(세부)"를 참조)

3. 개발내용

3-1. 일정표 및 주차별 세부 계획

추진 내용	수행기간(월) (계획표시 : ■)												비고	
	4 월				5 월				6 월					
	1	2	3	4	1	2	3	4	1	2	3	4		
데이터 수집 및 과제 설계	■													
데이터 분석을 위한 전처리 과정		■	■	■	■	■								
위험요소 분석 및 예측값 평가					■	■	■	■						
웹 프레임워크 구성							■	■	■	■	■	■		
웹 서비스 구현											■	■	■	

주	과제 추진 계획
1	과제 아이디어 도출을 위한 배경 조사
2	데이터 수집(공모전 참여, 중소기업청에서 데이터 제공)
3	수집한 데이터 분석, 과제 대략적 모습 설계
4	주요 요소(특징값) 도출, 가중치 부여
5	데이터 분석, 예측을 위한 가공(2주)
6	
7	위험요소 분석, 예측모듈 활용 : 예측값 평가(데이터 샘플)
8	예측값 평가(모든 데이터) 및 진행과정 검토(중간)
9	웹 프레임워크 구성을 위한 Django 기능 분석
10	UI(유저인터페이스) 및 기본기능 설계
11	시각화 및 예측자료와 코드 - 웹 페이지간 연동
12	기능 테스트 및 디버깅
13	웹 프레임워크 구성 (완료)
14	웹 서비스 구현(완료)
15	최종 결과보고

3-2. 구현방법(세부)

우선적으로 기본적인 시스템 설계를 진행하였음. (2-4 기본설계 부분)

구현하고자 할 웹 페이지 구성은 다음과 같음.

1. 종합 도산 가능성이 높은 상위 n개 기업을 막대그래프를 통해 시각화 및 보충설명
2. 업종별 도산 가능성이 높은 상위 n개 기업을 막대그래프를 통해 시각화 및 보충설명
3. 특정 코드(= 기업이름)를 가진 기업에 대한 도산 가능성 분석, 시각화

또한 파이썬을 사용하여 데이터 분석과 시각화하기 전 데이터를 가공(전처리)을 일부 수행하였음.

```

In [4]: 1 ## 1.3 데이터프레임 - 광주
        2 ## 2.4 데이터프레임 - 전남
        3
        4 import pandas as pan
        5 import numpy as np
        6
        7 df = pan.read_csv('경기전망19-2.csv', encoding = 'UTF-8')
        8
        9 GJ = df['X1'].isin([25]) # 지역코드(X1)가 25인 행들만 추려 GJ로
        10 JN = df['X1'].isin([36]) # 지역코드(X1)가 36인 행들만 추려 JN으로
        11
        12 df1 = df[GJ] # 지역코드가 25 : 광주의 기업만 추려내기 위한
        13 #df1_result = df1.sort_values(by = 'global_id', ascending = True, inplace = True) # 기업코드별 오름차순 정렬
        14 df3 = df1.set_index('global_id') # 기업코드 리스트로 빼내기 위한, 기업 리스트(기준) - 없는 기업의 경우 삭제
        15
        16 df2 = df[JN] # 지역코드가 36 : 전남의 기업만 추려내기 위한
        17 #df2_result = df2.sort_values(by = 'global_id', ascending = True, inplace = True) # 기업코드별 오름차순 정렬
        18 df4 = df2.set_index('global_id') # 기업코드 리스트로 빼내기 위한, 기업 리스트(기준) - 없는 기업의 경우 삭제
        19
        20 ref_GJ = [] # 앞으로 분석할 '광주'기업코드들의 기준 리스트 - 없거나 추가되는 기업이 나오기 때문에 연속성을 위해 리스트 만
        21 ref_JN = [] # 앞으로 분석할 '전남'기업코드들의 기준 리스트 - 없거나 추가되는 기업이 나오기 때문에 연속성을 위해 리스트 만
        22
        23 ref_GJ.append(df3.index)
        24 ref_JN.append(df4.index)
        25
        26 df3.head()
        27 #df4
        28
        29 # ref_GJ #128개 기업
        30 # ref_JN # 84개 기업
        31 # 총 192개 광주/전남 소재기업
        32 # 광주 / 전남 따로 분류하여 CSV파일로 저장
        33 # '9. 2월 ~ '20. 1월' 총 12개의 데이터를 12개 묶어서 분석
        34 # 만약 없는 행이 있을 경우에 ex) 2월에 존재하는 기업 코드 리스트를 가지고 불러올데
        35
        36 # 4월달에 해당 기업 데이터가 없을 경우에는
        37 # 결국지 3월/5월 데이터의 평균으로 각 값을 채운다
        38
        39
        40
    
```

Out [4]:

	X1	X2	X3	X16	X17	X17_1	X20	X21	X30	X31	...	X62	X63	X64	X65	X66	X67	X68	X69	X70	X71
global_id																					
134012	25	1	33	1	2	2	3	3	3	3	...	0	0	0	0	0	0	0	0	0	0
134013	25	3	30	1	2	2	3	3	3	3	...	0	0	0	0	0	0	0	0	0	0
134014	25	2	30	1	1	1	2	3	2	3	...	0	0	0	0	0	0	0	0	0	0
134016	25	3	95	1	2	2	2	3	0	0	...	0	0	0	0	0	0	0	0	0	0
134017	25	3	95	1	2	2	2	2	0	0	...	0	0	0	0	0	0	0	0	0	0

5 rows × 45 columns

[그림 5]. 데이터(.CSV) 중 일부를 불러와 가공하는 과정(코드)

그러나 과정 중 일부 문제가 발생하였는데, 바로 데이터들이 서로 정렬되어 있지 않고 다소 섞여 있어 한 번에 정리하기가 쉽지 않았음. 아래는 그 예시임

1. 특정 기업에 대한 데이터가 n월 데이터엔 존재하나 m월 데이터에는 존재하지 않음
2. 기업 코드 등 정렬되어 있지 않고 행의 갯수(= 기업의 갯수)가 맞지 않음

따라서 이러한 문제를 해결하기 위해 팀원이 모여 의견을 내놓았으며 그 결과는 다음과 같음.

1. **기준점 지정** : 기준 월을 지정하여 기업코드를 다른 월 데이터와 비교, 존재할 경우에만 분석
2. **카운트** : 기업코드를 카운트하여 기준 갯수 이상 존재할 경우 분석
(ex) 총 1년여 기간 데이터의 경우 n개 이상 존재할 경우에만 분석)

두 방식 모두 데이터 손실 문제가 발생할 가능성이 있기에 이러한 가능성을 최대한 줄이는 방향으로 해결 방법을 모색하였고, 2번째 방식을 보완하여 전처리를 진행하는 것으로 합의하였음. 즉 기업코드를 카운트하여 기준 갯수 이상 존재할 경우 분석하나, 기준 갯수의 크기를 낮춤으로써(ex) 10개 -> 7~8개만 존재해도 분석하도록) 데이터 손실 가능성을 최소화하고자 함.

항목번호	조사항목 내용	항목설명
x2	규모	1규모 ~ 6규모
x3	산업분류	
x16	기업유형	1: 일반기업 / 2: 벤처,이노비즈,경영혁신기업
x17	수출여부	1: 하고있다/2: 하고있지 않다
x17_1	대기업납품여부	1: 하고있다/2: 하고있지 않다
x20	경기실적	1: 매우나쁨 / 2: 다소나쁨 / 3: 동일 / 4: 다소좋음 / 5: 매우좋음
x21	경기전망	1: 매우나쁨 / 2: 다소나쁨 / 3: 동일 / 4: 다소좋음 / 5: 매우좋음
x30	생산실적(제조업)	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x31	생산전망(제조업)	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x32	내수실적	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x33	내수전망	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x34	수출실적	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x35	수출전망	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x36	영업이익실적	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x37	영업이익전망	1: 매우감소 / 2: 다소감소 / 3: 동일 / 4: 다소증가 / 5: 매우증가
x38	자금사정실적	1: 매우악화 / 2: 다소악화 / 3: 동일 / 4: 다소호전 / 5: 매우호전
x39	자금사정전망	1: 매우악화 / 2: 다소악화 / 3: 동일 / 4: 다소호전 / 5: 매우호전

[표 6]. 기업에 대한 재무상황을 나타내는 각종 지표(일부)

첫 번째로 전문가의 자문⁵⁾을 받아 기업에 대한 총 45개 평가 요소 중 10개의 주요 요소를 추출함. 전반적으로는 업종별, 내수/수출 중심 등 여러 기준에 따라 선택해야 할 요소가 다를 수 있으나 데이터셋에서 그러한 특징을 발견하기 쉽지 않았으므로, 모든 기업에 대해 포괄적으로 적용할 수 있는 요소를 선택하였고 그 결과는 다음과 같음.

항목번호	조사항목 내용	항목설명
global_id	-	조사단 통합 연계키
x1	지역	11: 서울 / 21: 부산 / 22: 대구 / 23: 인천 / 24: 광주 / 25: 대전 / 26: 울산 / 30: 경기북부 / 31: 경기남부 / 32: 강원 / 33: 충북 / 34: 충남 / 35: 전북 / 36: 전남 / 37: 경북 / 38: 경남 / 39: 제주
x2	규모	1규모 ~ 6규모
x3	산업분류	
x16	기업유형	1: 일반기업 / 2: 벤처,이노비즈,경영혁신기업
x17	수출여부	1: 하고있다/2: 하고있지 않다
x17_1	대기업납품여부	1: 하고있다/2: 하고있지 않다
x20	경기실적	1: 매우나쁨 / 2: 다소나쁨 / 3: 동일 / 4: 다소좋음 / 5: 매우좋음
x66	원자재(원재료)구매난	0: 미선택 / 1: 선택
x67	설비노후 및 부족	0: 미선택 / 1: 선택
x68	재정적비유기	0: 미선택 / 1: 선택
x69	판매통로	0: 미선택 / 1: 선택
x70	고급리	0: 미선택 / 1: 선택
x71	기업(대기업과의)물류경쟁력	0: 미선택 / 1: 선택

	항목번호	중요도
0. 기업코드	global_id	-
1. 자금사정실적	X38	25.0%
2. 내수전망	X33	15.0%
3. 판매대금회수지연	X57	11.0%
4. 자금조달곤란	X58	9.0%
5. 영업이익실적	X36	8.5%
6. 업체간과당경쟁	X59	7.5%
7. 경기전망	X21	7.0%
8. 인력확보난	X60	6.5%
9. 인건비상승	X61	6.0%
10. 수출전망	X35	3.5%

[그림 2]. 기업 평가에 대한 요소 일부(좌측), 주요요소 추출 및 가중치 부여(우측)

5) 임형열 : 친인척 중 은행업 종사자이신 분(IBK 중소기업은행 임직원)과 화상회의(통화)로 요소를 선정하였음.

모든 요소를 합산한 값을 1로 보았을 때 각 요소에 따른 중요도에 따라 가중치를 부여하였고, 이렇게 부여한 가중치와 선정 이유는 다음과 같음.

1. 자금사정실적(0.25) : 현재 기업의 자본금 보유 등 자본 사정이 가장 중요
2. 내수전망(0.15)⁶⁾ : 국내에서 활동하는 기업(내수중심)으로 가정
3. 판매대금회수지연(0.11) : 판매한 물품에 대한 대가 즉시 회수 가능 여부
4. 자금조달곤란(0.09) : 기업에 필요한 자금유통에 애로사항이 없어야 함
5. 영업이익실적(0.085) : 당연하지만, 기업의 영업이익률이 높을수록 좋음
6. 업체간과당경쟁(0.075) : 수요보다 공급이 많고 기술격차 적을 경우 경쟁 심화
7. 경기전망(0.07) : 저물어가는 산업 혹은 유망한 산업인지 여부 반영
8. 인력확보난(0.065) : 필요한 인력을 적시에 확보 가능한가 여부
9. 인건비상승(0.06) : 인력을 쉽게 확보 가능하더라도 인건비가 비쌀 경우
10. 수출전망(0.035) : 내수 중심 기업이라도 일정량의 수출은 필요

두 번째, '19년 2월부터 12월까지의 11개의 데이터셋에 대해 동일한 형태로 가공하는 과정을 진행하였음. 즉 본격적으로 데이터를 분석하고 예측모듈을 통해 결과값을 예측하는 과정을 수행하기 전 다소 형태가 복잡하고 무질서한 데이터의 형태를 모두 하나로 정렬하는 과정이 필요하였고, 이것을 세 명이 각자 나누어 시행함. 월별로 저장된 CSV 파일 형태의 데이터셋을 하나씩 열어 분석하였고 이후 공통부분을 찾아 코드를 자동화하여 분석할 수 있도록 하였음.

result9.set_index('월')												
	global_id	기업유형	경기전망	내수전망	수출전망	영업이익실적	자금사정실적	판매대금회수지연	자금조달곤란	업체간과당경쟁	인력확보난	인건비상승
월												
2	127117	2	2	2	0	3	2	0	1	1	0	0
2	134161	1	2	2	0	2	2	1	0	0	0	0
2	126988	1	3	2	0	2	2	0	1	0	1	1
2	134162	1	3	3	0	3	3	0	0	0	0	1
2	134163	1	3	2	0	2	3	0	0	1	0	1
...
12	118866	1	3	3	3	3	3	0	1	0	0	0
12	135633	1	3	3	0	4	4	0	1	1	0	0
12	135950	2	3	3	0	2	2	0	0	0	0	0
12	135921	1	3	3	0	3	3	0	0	1	0	0
12	135133	1	3	3	0	3	3	0	0	0	0	1

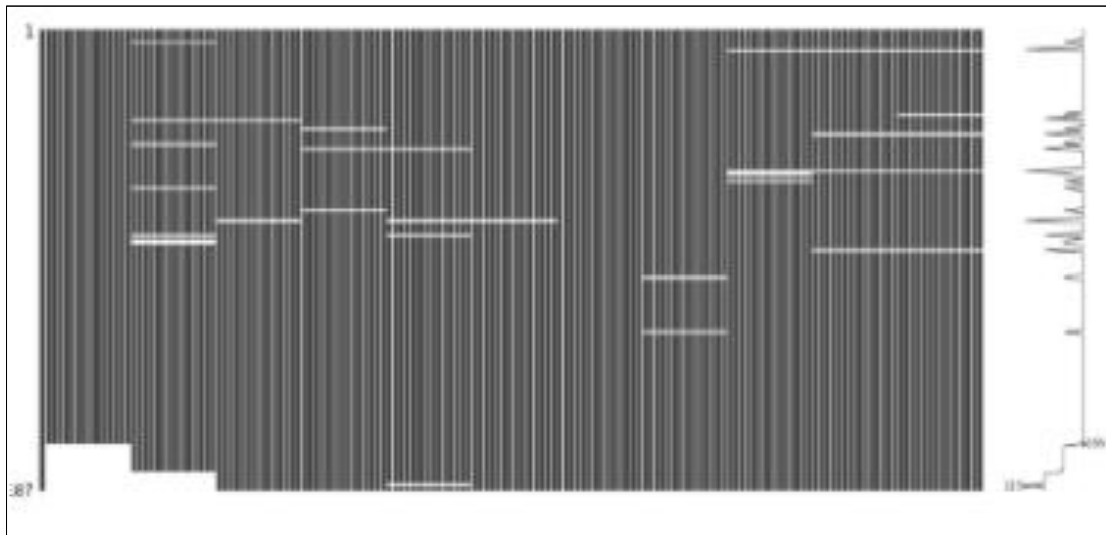
[그림 8]. 코드 자동화로 만든 데이터프레임 : 월을 index, 주요 요소를 특징으로 가짐

이전에 합의한 전처리 방식 -기업코드를 카운트하여 기준 갯수 이상 존재할 경우 분석- 으로 데이터를 가공하기 위해 월 별로 존재하는 기업코드를 각각 카운트하였고 그 결과 11개 데이터셋 중 값이 6개 이상 존재하는 기업이 **199개**, 8개 이상 존재하는 기업이 **197개**로 조건을 높였을 때 손실되는 데이터의 개수가 매우 적었으므로, 존재해야 할 값(기준)을 8개로 설정한 후 계속해서 데이터 가공을 진행하였음.

6) 수출 중심 기업의 경우 내수전망을 2번째로 중요한 요소로 선정하는 것은 맞지 않을 수 있음. 그러나 데이터셋에서 수출/내수 중심 기업을 분별할 수 없었고, 일반적으로 수출보다는 내수 중심 기업의 수가 더 많으므로 내수전망을 2번째 주요 요소로 선정함.

7) 김태완 : 각 데이터셋 분석 및 코드 자동화 가능한 공통부분 찾아냄
 김동진 : 공통부분을 활용한 코드 자동화(반복문 사용 모든 데이터를 가공하는 작업) 진행
 조재혁, 임형열 : 월별 존재하는 기업 개수 카운트, 공통부분을 활용한 코드 자동화 진행

또한 가공한 데이터에 결측치가 얼마나 남아있는지를 확인하기 위해, 파이썬의 별모 모듈(라이브러리) 'missingno'를 활용하여 결측 부분에 대한 시각화를 진행함.



[그림 9]. missingno 모듈을 활용한 결측치 부분 시각화(비어있는 부분이 결측치)

결측치를 어떤 방식으로 채워야 하는지에 대한 문제가 남아있어 이에 대해서 팀원들끼리 토의를 진행하여 해결 방법을 합의하였음. 토론 과정에서 크게 두 가지 의견이 나왔으며, 이에 대한 의견을 간단히 정리하자면 다음과 같음.

기 준	구체적 설명
분기	분기별 결측치가 존재할 경우 해당 분기의 평균을 이용
전월/익월	결측치를 가진 월의 전월/익월 평균을 이용

[표 1]. 결측치를 채우기 위한 기준과 그 방식(설명)

우리는 첫 번째 방식인 분기 기준을 선택하여 좀 더 개선된 방식을 사용하는 것으로 합의하였음. 즉, 각 분기마다 그룹화를 진행하여 결측치가 존재할 경우 해당 분기에 대한 평균값으로 대체하는 방식을 사용하기로 함.

이후 1. 예측모델의 신뢰성을 검증하기 위한 테스트셋을 생성, 데이터의 결측치 부분을 해결하는 과정과 2. 기업코드별 월간 데이터(행) 그룹화하여 재정렬하는 과정을 각각 진행하였으며 세부 설명은 아래와 같음.

1. 예측모델의 신뢰성을 검증하기 위한 테스트셋 생성

- 기존 진행방식은 우리가 가지고 있는 데이터 ('19년도 2~12월, '20년도 1월)중 '19년 데이터(11개월)만 데이터 전처리 과정을 통하여 예측모델을 생성, 이후 '20년 1월(1개월)에 해당하는 데이터와 비교하는 방법을 사용함. 그러나 '19년도 데이터 전처리 중 중소벤처기업부에서 임의로 부여한 global_id 항목이 규칙성이 다소 부족하여 정렬하기 어렵다는 사실을 발견함. 이를 보완하기 위해 global_id 항목을 cor_code : 규칙성 있는 숫자(지역코드+000n)로 변경함.([그림 1] 참고)
- 이 과정을 진행할 경우 기업명(코드)가 변경되어 '20년 데이터와 직접적인 비교하기 어려워짐. 따라서 이전에 작성하였던 코드(가공할 대상 데이터를 가져오는 부분)에 '20년 1월에 해당하는 데이터도 같이 가져와서 분석할 수 있도록 하였음.
- 이후 데이터를 동일한 기업코드를 가진 데이터를 월별로 정렬하는 부분에서 '20년 1월에 해당하는 데이터만 별도의 데이터프레임에 저장한 이후 예측모델 생성에 사용할 훈련 데이터에서는 삭제함.
- 예측모델의 신뢰성을 검증하는 가장 쉬운 방법은 실제로 있는 데이터와 직접 대조하는 것임. 따라서 본 팀은 '19년도 2월부터 12월까지 11개월간 데이터를 훈련 데이터로 사용, '20년 1월에 대한 예측모델을 생성한 후 실제로 존재하는 '20년 1월 데이터와 직접 대조해 봄으로써 신뢰성을 판단할 예정임(기존 신뢰도는 +/-10, 즉 80% 선으로 생각 중)
- 예측모델 생성에는 Facebook에서 개발한 fbprophet 라이브러리를 사용할 예정임.

경기전망	규모	기업유형	내수전망	수출전망	업체간 과당 경쟁	영업이익 실적	월	인건비상승	인력확보	자금사정	자금조달	지역	판매대금회수
cor_code													
240001	2.888889	5.000000	2.000000	2.222222	2.222222	0.333333	1.666667	7.555556	0.777778	0.000000	2.333333	0.222222	24.0
240002	3.222222	1.000000	1.000000	3.222222	2.222222	0.000000	2.777778	6.000000	0.666667	0.666667	2.888889	0.111111	24.0
240003	1.363636	4.000000	2.000000	1.363636	0.090909	0.909091	1.000000	7.000000	1.000000	0.636364	1.272727	1.000000	24.0
240004	2.727273	3.636364	1.000000	2.818182	0.000000	0.818182	2.909091	7.000000	0.181818	0.000000	3.000000	0.000000	24.0
240005	3.000000	1.000000	1.000000	3.000000	0.000000	0.090909	2.818182	7.000000	0.363636	0.000000	2.818182	0.090909	24.0
360006	1.333333	2.000000	1.000000	1.333333	0.000000	0.333333	1.222222	8.000000	0.444444	0.222222	1.222222	1.000000	36.0
360007	2.444444	2.000000	1.000000	2.444444	0.000000	0.666667	2.333333	8.000000	0.555556	0.111111	2.000000	0.888889	36.0
360008	2.111111	3.000000	1.000000	1.444444	0.000000	0.000000	1.222222	8.000000	0.777778	0.111111	1.111111	0.888889	36.0
360009	1.555556	6.000000	1.666667	1.555556	1.555556	0.333333	1.000000	8.000000	1.000000	0.222222	1.000000	1.000000	36.0
360070	3.222222	2.000000	1.000000	3.222222	0.000000	0.111111	3.000000	8.000000	0.888889	0.000000	3.111111	0.000000	36.0

[그림 1]. global_id를 cor_code로 변경 및 결측치 기입 : 소수형태로 값이 기입되는 문제)

2. 데이터의 결측치 부분 해결, 기업코드별 월간 데이터(행) 그룹화하여 재정렬

- 각 분기별로 그룹화하여 결측치를 채우는 방식을 사용한 후 소수로 결측치가 해결되는 문제점 발생, round()를 사용하여 소수점 첫째 자리에서 반올림하여 재기입하는 방식으로 해결.([그림 2] 참고)

기업유형	내수전망	수출전망	업체간 과당 경쟁	영업이익 실적	인건비상승	인력확보	자금사정	자금조달	지역	판매대금회수
cor_code										
240001	0	2.0	3.0	3.0	0.0	2.0	1.0	0.0	3.0	0.0 24.0
	0	2.0	2.0	2.0	0.0	2.0	1.0	0.0	2.0	0.0 24.0
	0	2.0	4.0	3.0	1.0	1.0	1.0	0.0	2.0	0.0 24.0
	0	2.0	2.0	2.0	0.0	2.0	1.0	0.0	2.0	0.0 24.0
360070	1.96	1.0	4.0	0.0	0.0	3.0	1.0	0.0	3.0	0.0 36.0
	1.96	1.0	3.0	0.0	0.0	4.0	1.0	0.0	4.0	0.0 36.0
	1.96	1.0	3.0	0.0	0.0	3.0	1.0	0.0	3.0	0.0 36.0
	1.96	1.0	3.0	0.0	0.0	2.0	0.0	0.0	3.0	0.0 36.0

[그림 1]. round() 함수를 사용. 채워진 결측치(소수)를 정수로 변환하여 채움.

8) FACEBOOK FBPROPHET : <https://facebook.github.io/prophet/>

이후 본 팀이 수행할 과정(예정사항)은 다음과 같음.

1. 자문 내용을 토대로 한 도산 가능성 산출 : 10가지 상위 요소와 각 가중치 기반.
2. fbprophet 라이브러리를 사용한 '20년도 1월에 대한 요소값 예측 및 검증 : 원 자료와 대조.
3. 산출된 도산 가능성과 요소값을 시계열 그래프로 시각화 : 분석 및 예측 결과.
4. 웹 프레임워크 Django를 사용한 웹 서비스 설계 및 구현 : 3번의 분석 및 예측 결과를 보여줌.

3-3. 협업방식 및 팀원 소감

본 팀은 개인 행동보다는 함께 상의하며 프로젝트를 진행하고 있음. 각 팀원마다 관심있고, 또 잘하는 분야가 달라서 이러한 부분을 반영하여 역할을 분담하였으나, 문제가 발생할 경우 자신의 역할과 분야가 아니더라도 해결을 위해 함께 고민하고 토론하며 진행하고 있음. 또한 팀원 모두가 잘 모르는 부분은 관련 서적과 인터넷 검색을 통해 비슷한 사례를 찾고 공유한 후 팀원들 간의 협의를 통해 프로젝트에 적용하고 있음. 특히 어려운 부분은 웹 프레임워크(Django)를 이용하여 웹 사이트 구축을 하는 부분이며, 팀원 모두 웹사이트 구축을 할 때 접해보지 못한 도구라서 함께 연구하고 있음. 아래는 팀원 별 소감임.

임형열 : 1학년부터 3학년까지 수강했던 과목들을 모두 마스터했다고 생각했는데 프로젝트를 진행하며 DB, Python 등에 대한 지식이 아직 부족하다는 점을 계속 느끼고 있습니다. 프로젝트가 끝난 뒤에도 부족하다고 느낀 분야에 대해 꾸준히 공부해야겠다는 생각이 들었습니다. 또한 부족한 분야를 팀원들이 많이 채워줘서 팀장으로서 미안하고 고마운 감정도 갖고 있습니다.

김동진 : 프로젝트를 진행하며 파이썬에 대한 관심이 더욱 증가했습니다. 굉장히 간편한 언어이며 문법도 간결해서 기존보다 흥미가 더 생겼습니다. 또 개인 역할에만 국한되지 않고 팀원 모두가 협업하는 방식을 통해 프로젝트를 진행함으로써 이전 학기에 진행했던 개인 프로젝트보다 부담감이 훨씬 덜했습니다.

조재혁 : 처음에 데이터를 선택 및 데이터 전처리 대해서 팀원과 같이 소통하며 의사결정을 하고, 문제점이나 에러가 발생하면 팀원들이 그 부분에 대해서 인터넷 검색이나 유사 자료를 찾아주어 하나씩 해결해주어 좋았고, 매주 문제점에 대한 결정사항을 발표 자료에 공유하면서 생각들을 정리해서 결정 사항에 대해 집중하여 해결할 수 있는 팀이라고 생각합니다.

김태완 : 이번에 파이썬과 장고 그리고 부트스트랩을 사용했었는데 처음 사용하다보니까 막히는 부분이 많이 있었습니다. 막힐 때마다 인터넷과 팀원들의 도움을 많이 받았습니다. 혼자하는 프로젝트였으면 많이 부담스러웠겠지만 팀원들이 많이 도와줘서 훨씬 부드럽게 진행이 되었습니다. 이번 기회를 통해서 많이 공부하여 프로젝트가 끝난 후에 실용적인 웹 사이트를 만들어 보고 싶습니다.

4. 부 록

4-1. 참고 서적과 관련 자료

본 프로젝트 설계와 구현에 참고한 자료는 다음과 같음. (최종보고까지 지속적으로 추가할 예정)

관련 서적

- 파이썬으로 데이터 주무르기 - 민형기 저, 비제이퍼블릭
- 파이썬 라이브러리를 활용한 데이터 분석 - 웨스 맥키니 외, 한빛미디어
- Django로 배우는 쉽고 빠른 웹 개발 - 파이썬 웹 프로그래밍 - 김석훈, 한빛미디어

Python External Module(Library)

- FACEBOOK FBPROPHET - <https://facebook.github.io/prophet/>
- PyPI DJANGO-CHARTS - <https://pypi.org/project/django-chartjs/>
- HIGHCHARTS - <https://www.highcharts.com/demo/>