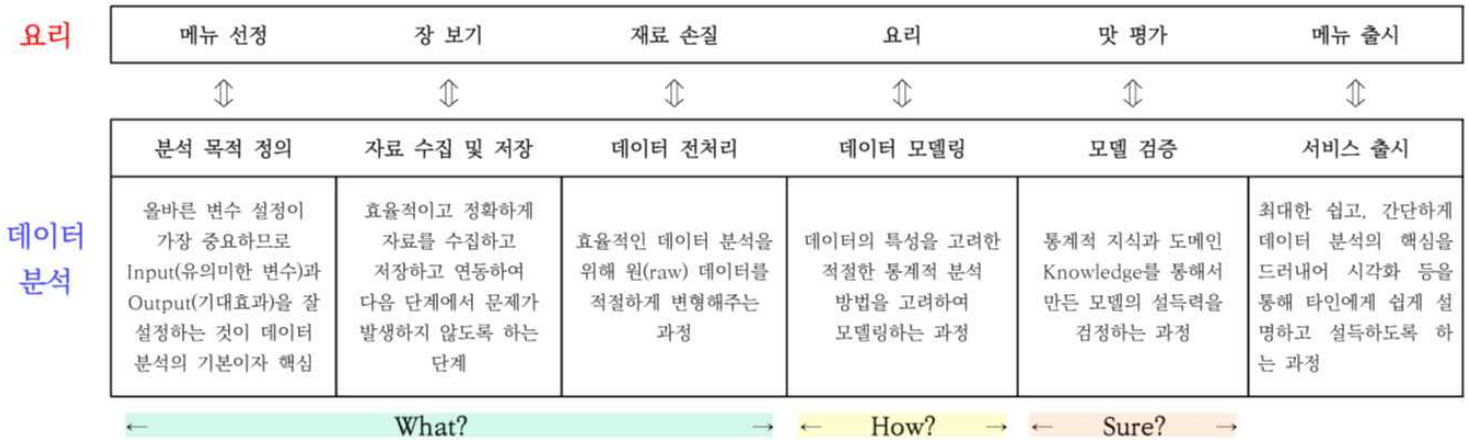
Data Analysis Process



단계(Step)			내용
1	Design	분석목적 정의	고객의 특성에 맞는 세분화된 서비스를 제공하는 효과적인 타겟 마케팅을 위해 다양한 고객층의 군집화
2	Preprocessing	자료 수집 및 저장	UCI Data set (CSV 자료)
3		데이터 파악 및 전처리, 가공	오타, 결측값, 데이터 단순화 등의 작업 요망
4	Modeling	적절한 분석 도구 개발	고객층을 분리하여 군집화하기 위한 효과적인 변수 결정을 위해 RFM 기반의 방법을 사용하여 K-평균 군집화
5	Validation	군집화 모델 평가	Silhouette Analysis(실루엣 분석)을 통해 도출되는 Silhouette Coefficient(실루엣 계수)값을 기반으로 성능 평가
6	Insight	평가, 결론, 시각화	군집화에 대해 잘 모르는 사람도 이해할 수 있도록 분석의 핵심을 쉽고 간결하게 요약 및 시각화

Step_3.1 데이터 파악 및 전처리

- 변수 의미 파악 -

InvoiceNo	StockCode	Description	Quantity	InvoiceDate
UnitPrice	CustomerID	Country		

주문번호	제품 고유 코드	상품 설명	주문 건수	주문 날짜
상품 단가	고객 번호	주문 국가명		

1단계: 총 541,909개의 데이터 존재하므로 현재 너무 큰 데이터의 볼륨을 줄이기 위하여 결측치, 오류, 제거 가능한 요소를 모두 제거하여 데이터를 가볍게 만든다.

① 총 135,080개의 Null(결측치) 존재하므로 이를 모두 제거
② Quantity 열에 총 10,624개의 음수항 존재. 주문 건수는 음수일 수 없으므로 해당 데이터는 오류(오타)로 판단하고 모두 제거
③ UnitPrice 열에 총 2개의 음수항 존재. 제품 단가 역시 음수일 수 없으므로 해당 데이터는 오류(오타)로 판단하고 모두 제거
④ Country 열에서 총 541,909 중 495,478개가 United Kingdom(영국) 즉, 전체의 약 90%가 같은 값을 갖는다. 분석의 편의를 위해 영국을 제외한 타 국가 모두 제거
⑤ 특정 고객이 주문 횟수와 주문 금액이 눈에 띄게 크다. 이 부분에 대해 고려해야 할 필요가 있다.

✔ ⑤를 제외한 위의 항목에 대하여 데이터 전처리를 실시하면 총 354,321개의 분석 가능한 데이터가 남는다.

2단계: CustomerID 열 전체가 명목형 변수 처리가 되어 있으므로 정수로 인식할 수 있도록 바꾼다.

Step_3.2 데이터 가공

- RFM 기법을 기반으로 한 데이터 가공 -

✔ 고객 군집화에 가장 많이 사용되는 기법인 'RFM 기법'은 Recency, Frequency, Monetary-value의 앞글자를 따서 만든 것으로 Recency는 고객이 마지막으로 제품을 구매한 후 오늘까지 지난 기간을, Frequency는 전체 상품 구매 횟수, Monetary-value는 전체 상품 구매 총액을 뜻한다. 한마디로 고객이 상품을 얼마나 자주, 많이 구매를 하는가에 따라 고객층을 분리한다는 기법으로 이해할 수 있다.

✔ 따라서 주어진 데이터의 변수의 의미와 형태를 파악하여 이를 Recency, Frequency, Monetary-value에 해당할 수 있도록 가공하는 것이 Step 2.0 단계에서의 핵심이다. 변수 가공 과정에 대한 세부 설명은 아래 표와 같다.

새로운 변수	기존 변수 가공 요소 및 과정
Recency	가장 최근 주문일자를 찾아내기 위해 CustomerID당 InvoiceDate 열(column)에서의 최대값을 구한다. 데이터는 2010.12.01~2011.12.09 기간 동안의 판매 자료이므로 계산의 기준이 되는 오늘은 2011.12.10.일 정도로 설정하도록 한다.
Frequency	CustomerID당 InvoiceNo의 개수가 곧 고객당 주문 건수를 의미한다.
Monetary-value	'UnitPrice'와 'Quantity'를 서로 곱하면 해당 주문시 총 주문 금액이 되므로 두 열(Column)을 곱한 후 CustomerID당 총 주문 금액을 계산한다.

★ 상단 표에 나온 변수 가공 과정은 모두 Python을 활용해 직접 코딩하여 실행하여 8주차에 '결과물 파일(html)'로 별도로 첨부 예정

① Unsupervised Learning(비지도학습)

: 데이터에 정해진 정답 라벨이 없는 데이터의 패턴이나 형태를 찾아내는 기계학습의 큰 종류 중 하나를 비지도학습이라고 한다. 비지도학습의 대표적인 종류로는 이번 분석에서 사용할 군집(Clustering)과 이외에 차원 축소(Dimensionality Reduction), 연관 규칙 학습(Association Rule Learning) 등이 있다.

② Clustering(군집분석)

: 집 분석은 각 개체의 유사성을 측정하여 높은 대상 집단을 분류하고, 군집에 속한 개체들의 유사성과 서로 다른 군집에 속한 개체 간 차이를 확인하는 분석으로 **이번 데이터 분석의 경우 고객의 구매 내역과 관련된 데이터를 통해 유사한 고객층과 상이한 고객층을 분리하여 각 군집에 맞는 타겟 마케팅을 효과적으로 할 수 있도록 하는 알고리즘을 만드는 것이 목표이므로 군집분석을 선택**했다.

③ K-Means Clustering(K-평균 군집분석)

: K-평균 군집분석은 군집분석 중에서도 비교적 쉽고 간결하여 많이 사용되는 알고리즘이다. **일반적인 방법이기도 하고, 조금 더 깊고 수준이 높은 분석을 진행할 시간적 여력이 없어 이 군집분석 방식을 선택**하였다. K-평균 군집분석은 원하는 만큼(K개)의 군집 또는 군집 중심점(centroid)을 초기값으로 지정하고, 해당 중심과 가까운 데이터를 선택하는 군집화 기법이다. 이때 각 군집의 평균을 시행마다 재계산하여 초기값을 갱신하며 이 알고리즘을 반복하게 되고 더 이상 중심점이 속한 군집이 바뀌지 않으면 군집화를 종료한다.

: 이론적으로 좀 더 살펴보자면, 사용되는 손실함수는 $\sum_{n=1}^{N}\sum_{n=1}^{K} I(x_n \in C_k) d(x_n, z_k)$이고, 이 손실함수를 최소로 하는 것을 목적으로 학습을 실시하는 것과 동일하며 반드시 수렴한다. k개의 군집 중심을 선택하는 과정에서 어떻게 선택하느냐에 따라 최종 결과가 달라지기 때문에 서로 다른 초기 군집 중심을 가지고 여러 번의 k-means 알고리즘을 수행한 후 가장 좋은 품질을 선택한다.

: K-평균 군집분석은 알고리즘이 비교적 쉽고 간단하지만 수행 시간이 느려지기 쉽고, 적절한 'K'값을 설정하는 것이 까다롭다는 단점이 있다. 또한 변수의 개수가 많을 경우 '차원의 저주'와 같은 문제점에 노출되므로 변수의 개수를 주링는 차원 축소 단계를 먼저 거쳐야 한다는 번거로움이 있다. 우리가 다루는 데이터의 경우 변수의 개수가 많지 않으므로 이론적인 차원 축소 과정은 생략하였다.

④ Silhouette Analysis(실루엣 분석)과 Silhouette Score(실루엣 스코어)

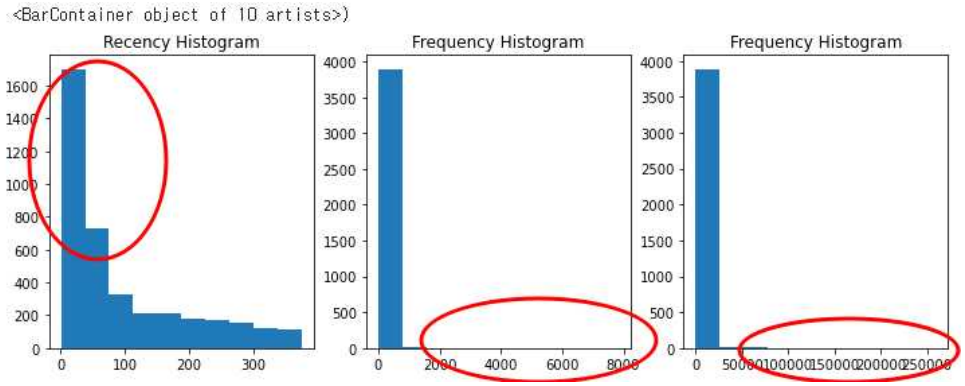
: 군집화가 효율적으로 잘 되었는가에 대한 평가 지표로서 Silhouette Analysis(실루엣 분석)의 Silhouette Coefficient(실루엣 계수)를 사용한다. 군집화된 결과물에서 군집 간의 거리가 잘 형성되어 있다는 것은 곧 군집화가 효율적으로 이루어졌다는 것을 의미한다. 따라서 실루엣 분석은 각 군집 간의 거리가 얼마나 효율적으로 분리되어 있는지 즉, 개별 군집들이 얼마나 비슷한 정도의 거리를 두고 떨어져 있는지를 판단해준다.

: 개별 데이터가 갖는 실루엣 계수는 해당 데이터가 같은 군집 내의 데이터와 얼마나 가깝게 군집화 되어 있고, 다른 군집의 데이터와는 얼마나 멀리 떨어져 있는가를 나타내는 지표다. 실루엣 계수는 −1과 +1 사이의 값을 가지도록 설정되어 있는데 값에 따른 의미는 다음과 같다.

✔ 실루엣 계수 값이 1에 가까워질수록 근처 군집과 멀리 떨어져 있다.
✔ 실루엣 계수 값이 0에 가까워질수록 근처 군집과 가까워진다.
✔ 실루엣 계수 값이 음수이면 전혀 다른 군집에 데이터가 할당되었다는 뜻이다.
✔ 좋은 군집화가 되기 위해서는 전체 데이터의 실루엣 계수에 대한 평균이 0과 1사이의 값을 가지면서, 1에 가까울수록 군집화가 효율적으로 되었다는 의미이며, 동시에 개별 군집의 평균값의 편차가 크지 않아야 한다.
✔ 실루엣 스코어가 모든 것을 의미하지는 않으며 데이터 시각화를 통해 실제 군집화가 균형있게 잘 이루어져 있는가를 확인해야 한다.

- 가공된 데이터를 기반으로 군집화 -

✔ 가공된 데이터를 군집화하기 전에 데이터의 분포를 파악하는 EDA 과정을 거쳐서 군집화 방향성에 대해 진단한다. 히스토그램을 통하여 가공된 새로운 데이터 셋(set)의 분포를 살펴보면 하단의 **<그림_1>**과 같이 특정 값에 심각하게 몰려 있는 점을 알 수 있었다. 수치적으로 확인할 수 있는데 **<그림_2>**를 보면 데이터의 평균이 중위값에 피해서 과도하게 높으로 최댓값 역시 상위 75%에 해당하는 값에 비해 압도적으로 높다는 것이 명백히 관찰된다.

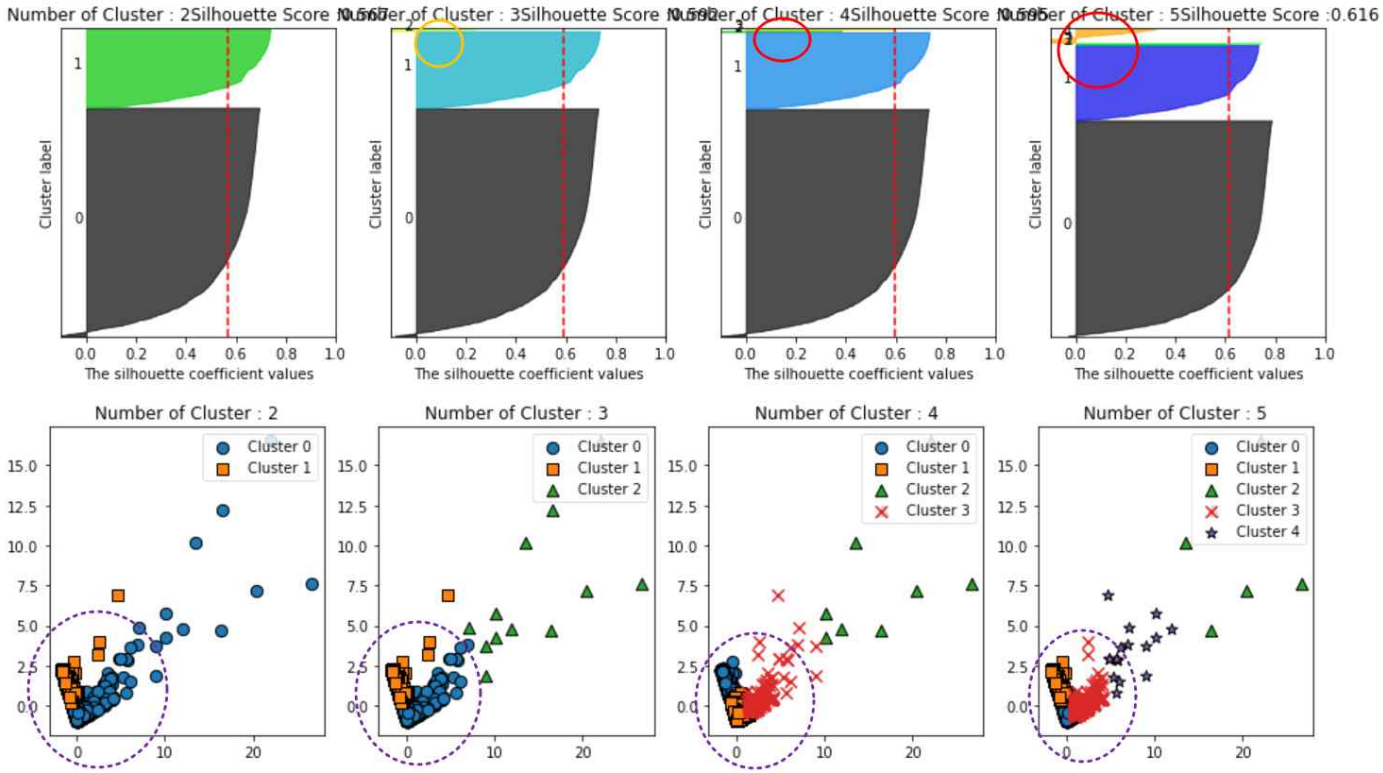


<그림_1>

	Recency	Frequency	Monetary
count	3920.000000	3920.000000	3920.000000
mean	92.742092	90.388010	1864.385601
std	99.533485	217.808385	7482.817477
min	1.000000	1.000000	3.750000
25%	18.000000	17.000000	300.280000
50%	51.000000	41.000000	652.280000
75%	143.000000	99.250000	1576.585000
max	374.000000	7847.000000	259657.300000

<그림_2>

✔ 실제로 위의 문제점을 분석 첫 단계에서 발견하지 못하고 그대로 군집분석을 실시했다. 그에 따라 'K'값에 따른 군집분석은 다음 페이지의 **<그림_3>**과 같은 결과가 나왔다.



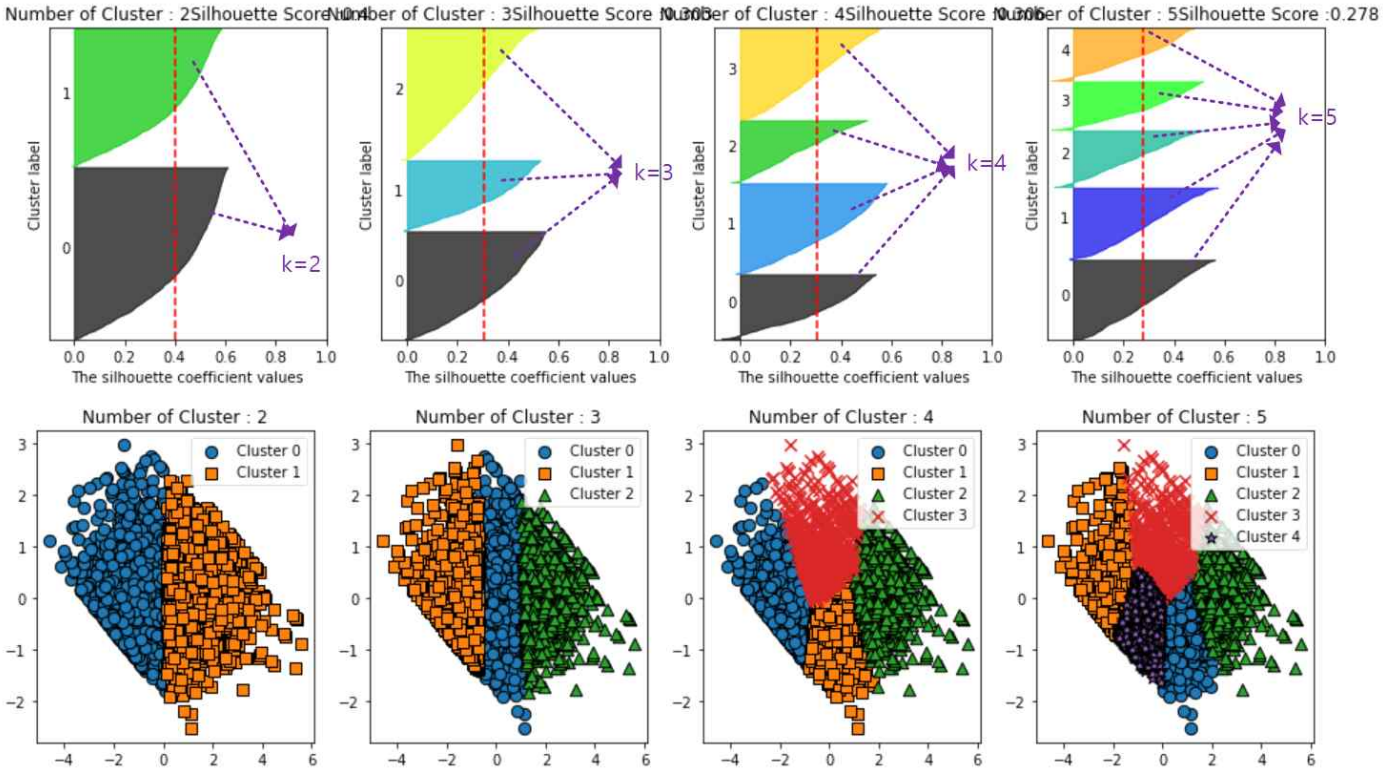
<그림_3>

✔ **<그림_3>**을 살펴보자. 위쪽 그림에서 색깔이 각각 다르게 색칠된 영역은 곧 서로 다른 군집을 의미한다. $k=2$일 때까지는 큰 문제가 없어보이나, $k=3$일 때 cluster 0(0번 군집)의 데이터 개수가 극도로 작으며, $k=4$와 $k=5$일 때에도 '특정 군집의 개수가 현저히 적다'는 유사한 형태의 문제점이 반복되었다. 또한 아래 그림을 관찰하면 같은 군집 내에서도 서로 근접하기보다는 너무 광범위하게 퍼져있어 같은 군집에 속한 데이터로 보기에 왜곡이 있다는 문제점이 발생했다. 위 그림에서는 글씨가 겹쳐 잘 보이지 않지만 $k=3$일 때 실루엣 스코어는 0.592로 나름 1에 가깝고 큰 문제가 없어보이지만 데이터 분포를 시각화 해 본 결과 문제점이 많이 발견되었다.

✔ 이러한 문제점이 발생하는 원인에 대해서 팀 전체가 토의해 본 결과 '데이터 파악 및 전처리 단계'에서 지적된 특정 고객의 구매 횟수와 구매 금액이 눈에 띄게 높다는 점이 문제였다는 결론이 나왔다. 이에 따라 데이터 왜곡 현상을 줄이기 위하여 '로그 변환(Log Transformation)'을 통해 전체적인 데이터 스케일링을 해준 후 다시 K-Means Clustering을 실시해봤다.

Step_5 & Step 6 수정된 K-Means Clustering 및 Visualization

- K-Means Clustering after 'Log Transformation' -



<그림_4>

✔ **<그림_4>**를 살펴보자. **<그림_3>**과는 확연히 다른 결과를 볼 수 있는데 설정한 군집 수(k)에 따라 군집이 균일하게 잘 나뉘었고, 군집 내의 데이터 역시 퍼져있지 않고 아름답게 밀집되어있는 모습을 볼 수 있다. 비록 실루엣 스코어 로그 변환 전보다 낮지만(글씨가 겹쳐 제대로 보이지 않음) 실루엣 스코어가 전부가 아니라 시각화를 통해 군집화를 확인 및 판단하는 과정이 상당히 중요하다는 것을 알 수 있었다.