# ADP 22회 리뷰

김승욱

0.기출문제요약
1.기계학습
2.통계분석

# 0.기출문제요약

## 1. 기계학습(머신러닝), (data: Pima Indian Diabetes)

- **탐색적 데이터 분석**: 결측치 확인, 히스토그램/박스플롯/페어플롯, 타겟변수 분포 그래프의 불균형 확인, 변수 전체의 상관관계, 이상치 처리 방안 제시, 위의 전처리 단계에서 얻은 향수 분석 시 고려사항 작성
- **클래스 불균형 처리**: 오버샘플링, 언더샘플링 과정 설명하고 결과 작성, 둘 중 선택하고 그 이유 설명
- **모델링**: 최소 3개 이상 알고리즘 제시하고 정확도 측면의 모델 1개와 속도 측면의 모델 1개를 구현, 모델 비교하고 결과 설명, 속도 개선을 위한 차원 축소 설명하고 수행, 성능과 속도 비교하여 결과 작성

## 2. 통계분석 (data: 금속 성분 함유량 데이터)

- 제품에 금속 재질 함유량의 분산이 1.3을 넘으면 불량이라고 보는데 제조사별로 차이가 남. 분산 검정 수행. 유의확률 0.05
- 불량률 관리도에 따른 관리 중심선, 관리 상한선, 하한선 구하기 (각 공식 있음), 관리도 시각화
- 표에 제품 1, 2를 만드는데 재료 a, b, c가 사용됨. 제품 1, 2는 각각 12만원, 18만원. 재료는 한정적일 때 최대 수익을 낼 수 있을 제품 1과 제품2의 개수 구하기
- 상품 a와 b가 있을 때 구매 패턴이 aa bb aaaa bbbb a b 등으로 나타날 때 두 상품의 연관성 유무를 검정할 것

# 0.기출문제요약

## 22회 기출 문제 분석

**1. 기계학습(50점)**

**1.데이터 탐색**(당뇨 데이터 세트, 데이터에 헤더가 없음, 대신 시험지에 변수명 제공)

1.1.1 탐색적 데이터 분석 수행하시오(시각화 포함)

1.1.2 이상치 처리하시오

1.1.3 앞선 두 단계에서 얻은 향후 분석시 고려사항 작성

**2. 클래스 불균형을 처리하시오**

2.1 업 샘플링 과정 설명하고 결과 작성 SMOTE

2.2 언더 샘플링 과정 설명하고 결과 작성

2.2 둘 중 선택하고 이유 설명

**3. 모델링 하시오**

3.1 최소 3개 이상 알고리즘 제시하고 정확도 측면의 모델 1개와 속도 측면의 모델 1개를 꼭 구현(총 2개 이상)

3.2 모델 비교하고 결과 설명

3.3 속도 개선을 위한 차원 축소 설명하고 수행, 예측 성능과 속도 비교하고 결과 작성

# 0.기출문제요약

## 22회 기출 문제 분석

**2. 통계분석 (50점)**

1. 금속 성분 함유량 변수 1개. (1열 데이터) 제품에 금속 재질 함유량의 분산이 1.3을 넘으면 불량이라고 보고 제조사별로 차이가 난다고 제보를 받는다. **분산에 대해 검정을 수행하시오.**

1.1 연구가설과 귀무가설 작성 / 1.2 양측 검정 / 1.3 검정통계량, 가설 채택

**2.Lot별 불량 제품 수량 데이터. lot 번호와 불량제품수 두 개의 열. 각 lot별 200개에 대한 불량제품 수.**

2.1 p관리도에 따라 관리중심선(center line), 관리 상한선, 하한선 구하시오

2.2 관리도 시각화 하시오

**3.표에 제품 1,2를 만드는데 사용되는 재료 a b c 컬럼 있고 재료에 따라 최종 만들어지는 제품 두 개에 대한 수량 있다. 제품 수량을 최대로 뽑으면서 수익이 최적이 되도록 작성하시오.(10점)**

**4. 상품 a와 b가 있을 때 다음과 같은 구매 패턴이 있다고 함. aa bb bbbb aa aaa bb bbb aa bb a b**

4.1구매하는 패턴으로 봐서 두 상품이 연관이 있는지 가설 세우고 검정하시오

4.2 연구가설 귀무가설 세우시오

4.3 가설 채택하시오

25:30

# *references

| context | url |
| --- | --- |
| 22회후기 전반 | https://lovelydiary.tistory.com/381 |
| 업/다운샘플링 | https://min23th.tistory.com/19<br>https://codedragon.tistory.com/9861<br>https://thebook.io/006723/ch10/06/01/ |
| 인디언 당뇨병 환자 데이터 | https://www.kaggle.com/uciml/pima-indians-diabetes-database/version/1 |
| Confusion matrix | https://blog.naver.com/sjy5448/222458248562 |
| Run test | https://blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=li0224il&logNo=220722414973 |
| 데싸라면 세미나(21.12.30) | 제작자가 직접 공유한 영상 캡쳐 |

# 1.기계학습(50점)

**1.1.1 탐색적 데이터 분석 수행(시각화 포함)**

-dim(), str(), summary() 기본 함수
-결측치 확인
-독립변수 전체 히스토그램+박스플랏+pairplot
-타겟변수 분포 그래프(불균형 확인)
-변수 전체 상관관계

@인디언 당뇨병 데이터 from kaggle

```
> diab = read.csv("diabetes.csv")
> dim(diab)
[1] 768   9
> summary(dim)
Error in object[[i]] : object of type 'builtin' is not subsettable
> summary(diab)
  Pregnancies        Glucose      BloodPressure    SkinThickness      Insulin          BMI        DiabetesPedigreeFunction      Age          Outcome
 Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00   Min.   :  0.0   Min.   : 0.00   Min.   :0.0780           Min.   :21.00   Min.   :0.000
 1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00   1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437           1st Qu.:24.00   1st Qu.:0.000
 Median : 3.000   Median :117.0   Median : 72.00   Median :23.00   Median : 30.5   Median :32.00   Median :0.3725           Median :29.00   Median :0.000
 Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54   Mean   : 79.8   Mean   :31.99   Mean   :0.4719           Mean   :33.24   Mean   :0.349
 3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00   3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262           3rd Qu.:41.00   3rd Qu.:1.000
 Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00   Max.   :846.0   Max.   :67.10   Max.   :2.4200           Max.   :81.00   Max.   :1.000
> str(diab)
'data.frame':   768 obs. of  9 variables:
 $ Pregnancies             : int  6 1 8 1 0 5 3 10 2 8 ...
 $ Glucose                 : int  148 85 183 89 137 116 78 115 197 125 ...
 $ BloodPressure           : int  72 66 64 66 40 74 50 0 70 96 ...
 $ SkinThickness           : int  35 29 0 23 35 0 32 0 45 0 ...
 $ Insulin                 : int  0 0 0 94 168 0 88 0 543 0 ...
 $ BMI                     : num  33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...
 $ DiabetesPedigreeFunction: num  0.627 0.351 0.672 0.167 2.288 ...
 $ Age                     : int  50 31 32 21 33 30 26 29 53 54 ...
 $ Outcome                 : int  1 0 1 0 1 0 1 0 1 1 ...
> library(caret)
```

# 1.기계학습(50점)

**1.1.2 이상치 처리(이상값 대체방안 제시)**

-참고: EDA문제에서 이상치를 확인하여 처리하는것보단 이번단계에서 처리할껄.

- 독립변수 summary()와 히스토그램+pairplot을 통해 이상치 식별가능

*의도적인 9999값들이 있었으나 케글데이터에는 없음

- 일부 독립변수의 경우 값들이 이상치를 보여서 상하한선 5% 기준으로 이상치 대체를 진행

(단, 나이/임신횟수는 실제정보인관계로 제외함)

일반 적으로 IQR을 사용하지만 데이터에 큰 변환을 줄수있으므로 상하한5%기준으로 이상치 대체를 사용

```
> summary(diab)
  Pregnancies        Glucose      BloodPressure    SkinThickness      Insulin          BMI        DiabetesPedigreeFunction      Age          Outcome
 Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00   Min.   :  0.0   Min.   : 0.00   Min.   :0.0780           Min.   :21.00   Min.   :0.000
 1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00   1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437           1st Qu.:24.00   1st Qu.:0.000
 Median : 3.000   Median :117.0   Median : 72.00   Median :23.00   Median : 30.5   Median :32.00   Median :0.3725           Median :29.00   Median :0.000
 Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54   Mean   : 79.8   Mean   :31.99   Mean   :0.4719           Mean   :33.24   Mean   :0.349
 3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00   3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262           3rd Qu.:41.00   3rd Qu.:1.000
 Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00   Max.   :846.0   Max.   :67.10   Max.   :2.4200           Max.   :81.00   Max.   :1.000
> summary(diab$BloodPressure)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   62.00   72.00   69.11   80.00  122.00
> dim(diab)
[1] 768   9
> str(diab)
'data.frame':   768 obs. of  9 variables:
 $ Pregnancies             : int  6 1 8 1 0 5 3 10 2 8 ...
 $ Glucose                 : int  148 85 183 89 137 116 78 115 197 125 ...
 $ BloodPressure           : int  72 66 64 66 40 74 50 0 70 96 ...
 $ SkinThickness           : int  35 29 0 23 35 0 32 0 45 0 ...
 $ Insulin                 : int  0 0 0 94 168 0 88 0 543 0 ...
 $ BMI                     : num  33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...
 $ DiabetesPedigreeFunction: num  0.627 0.351 0.672 0.167 2.288 ...
 $ Age                     : int  50 31 32 21 33 30 26 29 53 54 ...
 $ Outcome                 : int  1 0 1 0 1 0 1 0 1 1 ...
>
```

# 1.기계학습(50점)

**1.1.3 앞선 두 단계에서 얻은 향후 분석시 고려사항 작성**
결측치가 없는 것을 확인 후 이상치 대체를 진행하였다고 언급.

또한 타겟변수 분포가 불균형(imbalance)하다는 것을 언급하며 이는 분석함에 있어 유의해야할 부분이며, 타겟변수 분포(0,1)별 독립변수에 대한 차이가 있는지 확인해야한다고 하였습니다.
(아래에 불균형 관련으로 문제가 나와서) 2:1정도로 차이가 나니 up-sampling 또는 under-sampling을 해야할수도 있다고 하였습니다.

# 1.기계학습(50점)

## 2.1 클래스 불균형 처리

@caret 패키지의
upSample(),
downSample() 함수

### 2.1.1 오버샘플링 과정 설명하고 결과 작성

*upSample() 함수 사용. 업샘플링 전후 dim(), summary() 아웃풋값 비교.

### 2.2.2 언더샘플링 과정 설명하고 결과 작성

*downSample() 함수 사용.언더샘플링 전후 dim(), summary() 아웃풋값 비교.

### 2.2.3 둘 중 선택하고 이유 설명

둘 중 선택한다면 *언더샘플링을 선택*한다고 하였습니다.
1) 업샘플링의 경우 데이터가 적은 쪽을 표본으로 더 많이 추출하면서 정보 손실은 없지만 *중복 관측치가 생기게 되어 오버 피팅*에 대한 가능성이 있습니다.

2)다운샘플링을 하게 된다면 데이터가 많은 쪽을 적게 추출함으로 *정보 손실*이 있을 수 있지만, 다운샘플링 전후의 시각화 및 summary 값을 비교하여 큰 차이가 나지 않아 다운샘플링을 진행해도 괜찮을 것 같다고 판단하였습니다.

*데이터셋의 경우에는 굳이 중복 관측치를 만들면서까지 업샘플링을 하지않고 언더샘플링을 선택하겠다고 서술함.

*업샘플링 : 해당 분류에 속하는 데이터가 적은 쪽을 표본으로 더 많이 추출하는 방법

*언더샘플링 : 해당 분류에 속하는 데이터가 많은 쪽을 적게 추출하는 방법

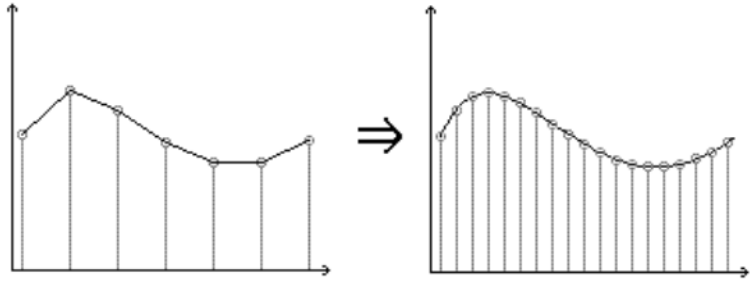
# *업/다운 샘플링

업 샘플링 : 해당 분류에 속하는 데이터가 적은 쪽을 표본으로 더 많이 추출하는 방법
다운 샘플링: 해당 분류에 속하는 데이터가 많은 쪽을 적게 추출하는 방법

ex) 총 6개의 데이터가 있고 각각의 분류가 '0, 0, 0, 0, 1, 1'이라고 하자.
이 데이터를 그대로 쓰면 0을 예측할 확률이 큰 모델이 만들어지므로, 그대로 쓰지 않고 표본을 추출한다.
복원 추출 방식으로 분류 0과 1에서 각각 표본을 4개씩 뽑으면 **'0, 0, 0, 0, 1, 1, 1, 1'**이 되어 각 분류에 해당하는 데이터의 개수가 같아진다. 이것이 **업 샘플링 방법**이다.
반대로 다운 샘플링에서는 총 2개씩의 데이터를 0과 1에서 표본으로 뽑아 '0, 0, 1, 1'이 된다.

업 샘플링(up sampling) & 다운 샘플링(down sampling)

| 구분 | 설명 |
| --- | --- |
| 업 샘플링 (up sampling) | • 해당 분류에 속하는 데이터가 적은 쪽을 표본으로 더 많이 추출하는 방법입니다.<br>• R: 패키지의 upSample()함수를 통해 업 샘플링 방법을 수행할 수 있습니다. |
| 다운 샘플링 (down sampling) | • 해당 분류에 속하는 데이터가 많은 쪽을 적게 추출하는 방법입니다.<br>• R: 패키지의 downSample()함수를 통해 다운샘플링 방법을 수행할 수 있습니다. |



출처 : https://t1.daumcdn.net/cfile/tistory/26502E3357B1CCE22C

업샘플링

# 1.기계학습(50점)

**3.1 모델링**

**3.1.1 최소 3개 이상 알고리즘 제시하고**
**정확도 측면의 모델 1개와 속도 측면의 모델 1개를 꼭 구현(총 2개 이상)**

- **아쉬운 부분** : 알고리즘보다 모델 위주로 설명을 적어서 이부분에서 감점을 받을 것 같다는 생각이 듭니다.
- 만약 적는다면 선형/배깅/부스팅 알고리즘을 제시하여 부가적인 설명을 적을 것 같습니다.

3개 알고리즘 : Logistic Regression, Random Forest, XGBoost
정확도 측면의 모델 1개 : Random Forest
속도 측면의 모델 1개 : Logistic Regression

@제가 작성한 답중에
정확도 랜덤포레스트. 속도 의사결정나무

# 1.기계학습(50점)

**3.1.2 모델 비교하고 결과 설명**

**아쉬운 부분 1** : 앞의 문제에서 언더샘플링을 해준만큼 해당 부분을 조금 더 언급하거나, 데이터와 독립변수가 추가적으로 더 수집되면 더욱 비교를 잘할 수 있을 것 같다 등과 같이 언급하면 좋지 않았을까 생각합니다.

**아쉬운 부분 2** : 데이터수가 많지 않은만큼 k-fold cross validation을 사용하여 모델별 mean accuracy를 산출해서 비교하는 것이 더 정확하지 않았을까 생각합니다.

*속도를 계산할 때는 Sys.time() 함수를 사용하였습니다.
해당 함수는 시스템 함수로 코드 청크의 시작과 끝의 시간 차이를 측정해줍니다.

결과 : 정확도 측면에서는 Random Forest, 속도 측면에서는 Logistic Regression이 좋았습니다.
Confusion Matrix 값을 사용하여 전체적인 평가 지표를 산출 및 제공하였습니다.

```
> Sys.time()
[1] "2022-01-16 01:36:14 KST"
> 
```

# *confusion matrix

#예측을 통한 정분류율 확인
> library(caret)
> pred.nn<-predict(nn.model, test[,-1], type="class")
> confusionMatrix(data=as.factor(pred.nn),reference=test[,1], positive='1')
Confusion Matrix and Statistics

Reference
Prediction 0 1
0 14 5
1 87 194

Accuracy : 0.6933
95% CI : (0.6378, 0.745)
No Information Rate : 0.6633
P-Value [Acc > NIR] : 0.1494

Kappa : 0.1418

Mcnemar's Test P-Value : <2e-16

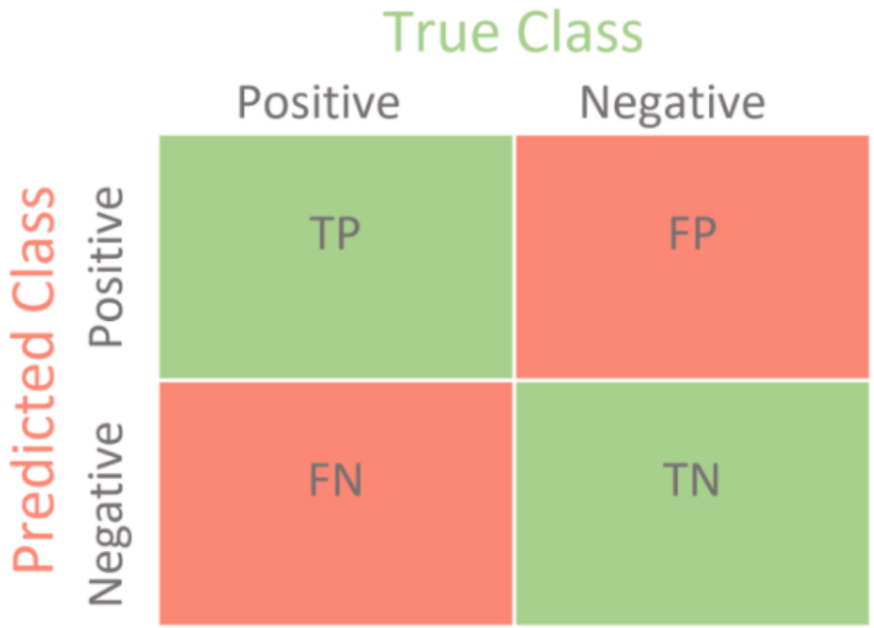
Sensitivity : 0.9749
Specificity : 0.1386
Pos Pred Value : 0.6904
Neg Pred Value : 0.7368
Prevalence : 0.6633
Detection Rate : 0.6467
Detection Prevalence : 0.9367
Balanced Accuracy : 0.5567

'Positive' Class : 1

# *confusion matrix

## 오차 행렬(Confusion Matrix)

오차 행렬(Confusion Matrix, 혼동행렬)은 학습된 분류 모델이 예측을 수행하면서 얼마나 헷갈리고 (confused) 있는지도 함께 보여주는 지표이다. 즉, 이진 분류의 예측 오류가 얼마인지와 더불어 어떠한 유형의 예측 오류가 발생하고 있는지를 함께 나타내는 지표이다.




오차 행렬은 이러한 4분면 행렬에서 실제 레이블 클래스 값과 예측 레이블 클래스 값이 어떠한 유형을 가지고 매핑되는지를 나타낸다. 4분면 행렬에 있는 TP, FP, FN, TN 값을 다양하게 결합해 분류 모델 예측 성능의 오류가 어떠한 모습으로 발생하는지 알 수 있는 것이다. TP, FP, FN, TN 기호가 의미하는 것은 앞 문자 True/False는 예측값과 실제 값이 '같은가/틀린가'를 의미한다. 뒤 문자 Negative/Positive는 예측 결괏값이 부정(0)/긍정(1)을 의미한다.

TN : 예측값을 Negative(0)로 예측했고 실제 값 역시 Negative(0)

FP : 예측값을 Positive(1)로 예측했고 실제 값은 Negative(0)

FN : 예측값을 Negative(0)로 예측했고 실제 값은 Positive(1)

TP : 예측값을 Positive(1)로 예측했고 실제 값 역시 Positive(1)

이 값들을 조합해 Classifier의 성능을 측정하는 주요 지표인 정확도(Accuracy), 정밀도(Precision), 재현율(Recall) 값을 알 수 있다. 앞서 정확도를 이야기하면서 언급한 추가적인 계산식 또한 이 값을 조합한다.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

정확도(Accuracy)의 계산식

일반적으로 불균형한 레이블 클래스를 가지는 이진 분류 모델에서는 많은 데이터 중에서 중점적으로 찾아야 하는 매우 적은 수의 결괏값(ex> 사기 행위 예측 모델에서 사기 행위)에 Positive를 설정해 1값을 부여하고, 그렇지 않은 대부분의 경우에 Negative로 0 값을 부여하는 경우가 많다. 따라서 정확도 지표는 비대칭한 데이터 세트에서 Positive에 대한 예측 정확도를 판단하지 못한 채 Negative에 대한 예측 정확도만으로도 TN의 값이 매우 커지기 때문에 분류의 정확도가 매우 높게 나타나는 수치적인 판단 오류를 일으키게 된다.

# 1.기계학습(50점)

**3.1.3 속도 개선을 위한 차원 축소 설명하고 수행, 예측 성능과 속도 비교하고 결과 작성**

**아쉬운 부분** : 표준화 및 주성분분석을 진행 시 test leakage를 조심해야하는데, 이 부분을 간과하고 진행한듯한 기억이 납니다
이 부분은 다시 공부하여 내용 정리하는 글을 올릴 예정입니다.

주성분분석(PCA)를 사용하였으며, 해당 개념을 설명하였습니다.
scaling을 진행하였고, scale() 함수를 사용하였습니다.

설명 분산 누적은 85%로 설정하였고, 요약된 주성분이 전체 데이터의 85%정도를 설명하였습니다. (Comp4 정도까지)

속도와 예측 성능이 둘다 다소 하락한 것을 확인할 수 있었습니다. 만약 데이터와 독립 변수가 추가적으로 더 있다면 PCA로 진행했을 때, 속도가 더욱 하락하고 예측 성능도 어느정도 높게 나올 수 있을 것이라고 서술했습니다.

# *주성분분석

변수 축소 --분석에 사용할 변수가 많을경우 주성분분석/요인분석 사용하여 변수축소한다.

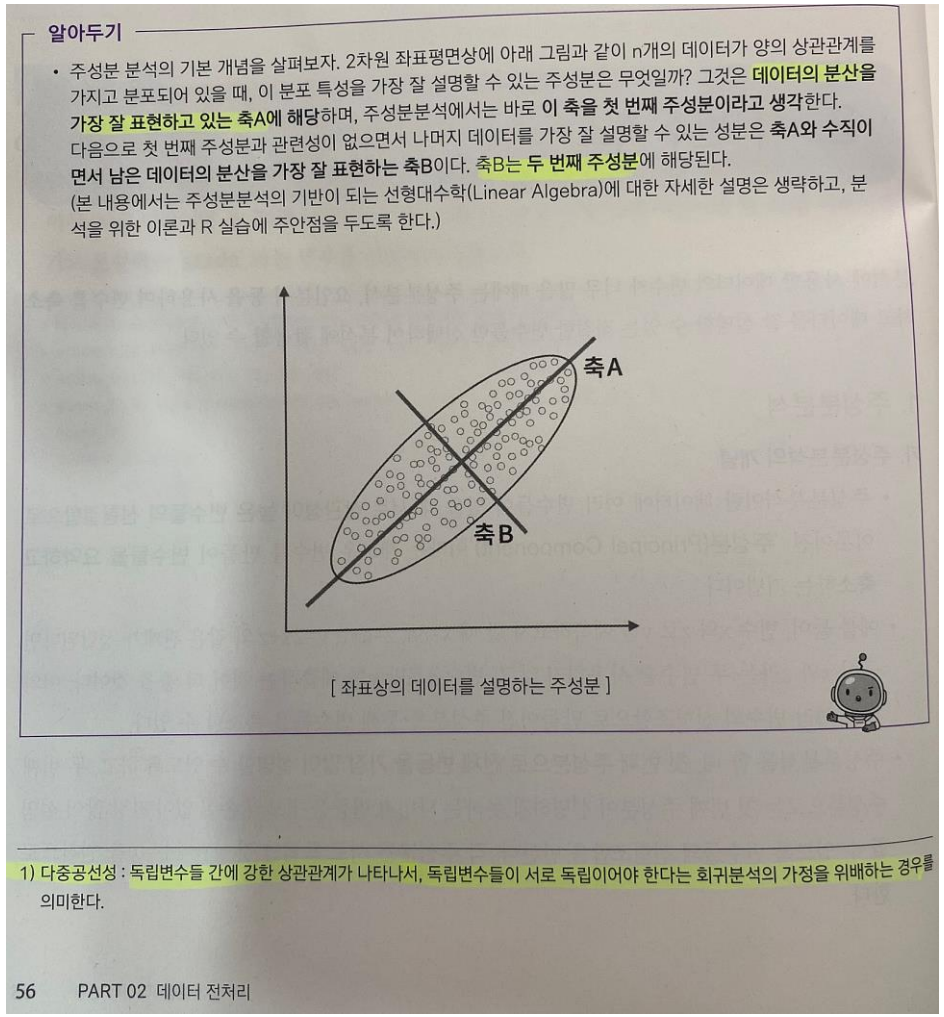
## 1. 주성분분석

### 가.주성분분석의 개념

- 데이터에 여러 변수들이 있을 때 **서로 상관성이 높은 변수들의 선형결합**으로 이루어진 **'주성분'**이라는 새로운 변수를 만들어 **변수들을 요약하고 축소하는 기법.**

*주성분: Principal Component

- 첫 번째 주성분으로 전체 변동을 가장 많이 설명할수있고, 두번째 주성분으로는 첫번째 주성분이 설명하지 못하는 나머지 변동을 정보의 손실 없이 가장 많이 설명할 수 있도록 변수들의 선형조합을 만든다. **각 주성분은 서로 독립**인 것(상관계수=0)을 원칙으로 한다.



알아두기

• 주성분 분석의 기본 개념을 살펴보자. 2차원 좌표평면상에 아래 그림과 같이 n개의 데이터가 양의 상관관계를 가지고 분포되어 있을 때, 이 분포 특성을 가장 잘 설명할 수 있는 주성분은 무엇일까? 그것은 **데이터의 분산을 가장 잘 표현하고 있는 축A에 해당**하며, 주성분분석에서는 바로 **이 축을 첫 번째 주성분이라고 생각**한다. 다음으로 첫 번째 주성분과 관련성이 없으면서 나머지 데이터를 가장 잘 설명할 수 있는 성분은 **축A와 수직이면서 남은 데이터의 분산을 가장 잘 표현하는 축B**이다. 축B는 **두 번째 주성분**에 해당된다.
(본 내용에서는 주성분분석의 기반이 되는 선형대수학(Linear Algebra)에 대한 자세한 설명은 생략하고, 분석을 위한 이론과 R 실습에 주안점을 두도록 한다.)



[ 좌표상의 데이터를 설명하는 주성분 ]

1) 다중공선성 : 독립변수들 간에 강한 상관관계가 나타나서, 독립변수들이 서로 독립이어야 한다는 회귀분석의 가정을 위배하는 경우를 의미한다.

# *주성분분석

**나.주성분분석의 목적**
- 변수들 간에 내제하는 상관관계, 연관성을 이용해 **소수의 주성분으로 차원을 축소.**
- **다중공선성**이 존재하는 경우, **상관성이 없는(적은) 주성분으로 변수들을 축소**하여 모형 개발에 활용할 수 있다.
(회귀분석이나 의사결정나무 등의 모형 개발 시 입력변수들 간의 상관관계가 높은 **다중공선성(multicollinearity)**이 존재할 경우 모형이 잘못 만들어져 문제생김)
- 주성분분석을 통해 변수차원을 축소한후 **군집분석을 수행하면 군집화 결과와 연산속도를 개선** 할수 있다.

*다중공선성: 독립변수들 간에 강한 상관관계가 나타나서, 독립변수들이 서로 독립이어야 한다는 회귀분석의 가정을 위배하는 경우.
#알아두기
데이터의 분산을 가장잘표현하고있는 축A, 이축을 첫번째 주성분.
다음으로 축A와 수직이면서 남은데이터의 분산을 가장 잘 표현하는 축B는 두번째 주성분.

# 2.통계분석(50점)

데싸라면 풀이
스토리텔링을 통해 어떤 검정을 해야하는지 묻는
ANOVA 검정(분산에 대한 검정)
독립변수 = 제조사(변수형)
종속변수 = 금속재질함유량(연속형)

> 총 4개의 문제가 있었으며, '산공과 교수님이 냈나?'할 정도로 품질경영 문제가 나와서 당황스러웠습니다😀

**1. 금속 성분 함유량 데이터(변수 1개) - 제품에 금속 재질 함유량의 분산이 1.3을 넘으면 불량이라고 보고 있는데 제조사별로 차이가 난다고 제보를 받았으며, 분산에 대해 검정을 수행하시오. (유의확률 0.05)**

- **아쉬운 부분 1** : 아무리 기억을 복기해도 분산을 제공했던 기억은 없는데, 문제를 해석할 때 위와 같이 해석을 해야했던 것 같습니다. 아니면 제가 맘이 급해서 못 봤을수도 있어요😭
- **아쉬운 부분 2** : 일표본 카이검정(One Sample Chi-square Test)을 해야하는건 파악했는데, 어떻게 진행해야하는지 코드를 몰라서 진행하지 못했습니다.

**1.1 연구가설과 귀무가설 작성**

- $H_0 : \sigma^2 = \sigma_0^2 H_0 : \sigma^2 = \sigma_0^2$

# 2.통계분석(50점)

### 1.2 양측 검정

- **삼질 대잔치** : 등분산성 검정을 해야한다고 판단하여 `var.test()`를 사용할 때 x는 데이터가 들어가는데 y에는 vector가 들어가야하는데 어떻게 해야하는지 버벅거렸습니다. 그냥 1.3 분산 가지는걸로 임의로 만들어서 비교하면 되지 않았을까 생각했는데, 일표본 카이 검정이니 이건 아니었겠죠..? 에효😭
  - `var.test(x, y, ratio= 1, alternative=c("two.sided","less","greater"), conf.level=0.95)`

- **정답** : 분산을 추정하고 분산이 사용자 지정 값과 같다는 카이 제곱 검정을 사용하여 귀무 가설 검정 및 분산에 대한 신뢰 구간을 구하는 함수가 있었습니다. EnvStats 패키지의 `varTest` 함수를 사용하는게 제일 베스트였던 것 같습니다.
  - varTest: One-Sample Chi-Squared Test on Variance
  - `varTest(x, alternative = "two.sided", conf.level = 0.95, sigma.squared = 1, data.name = NULL)` 에서 sigma.squared = 1.3으로 지정하여 진행하면 됩니다.

### 1.3 검정통계량, 가설 채택

- 검정 통계량 : $\chi_0^2 = \frac{(n-1)S^2}{\sigma_0^2}$ $\chi_0^2 = \frac{(n-1)S^2}{\sigma_0^2}$

# 2.통계분석(50점)

데싸라면 풀이
평균값(금속재질함유량)에서 분산1.3이 넘어가는것을 상하한을 시각화 해주면됨.

관리도?
관리 중심선 = 평균, 관리 상한선과 하한선은 *평균 기준 3표준편차씩 거리에 위치한 선*

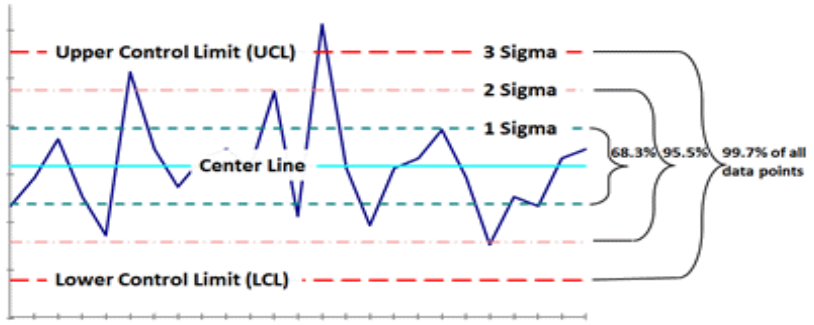
**2. Lot별 200개에 대한 불량 제품 수량 데이터(변수 2개 - lot번호, 불량제품수)**

**2.1 불량률 관리도에 따라 관리중심선(CL : Center Line), 관리 상한선(UCL : Upper Control Limit), 하한선(LCL : Lower Control Limit) 구하기**

- 불량률에 대한 관리도이기에, 먼저 불량률을 산출해주었습니다. ( $\frac{불량제품수*100}{200}$ )
- 관리중심선(CL)은 평균값이고, 관리 상한선(UCL)과 관리 하한선(LCL)은 평균으로부터 $3\sigma 3\sigma$(표준편차) 떨어진 값입니다.
  - 관리 상한선(UCL) : $\mu + 3\sigma \mu + 3\sigma$
  - 관리 하한선(LCL) : $\mu - 3\sigma \mu - 3\sigma$

**2.2 관리도 시각화**

- 시각화는 구글에 python control chart 검색하면 많이 나오고, 결국 관리도가 생소한 말이지만, 관리 중심선 = 평균, 관리 상한선과 하한선은 평균 기준 3표준편차씩 거리에 위치한 선입니다.
- 관리도 시각화는 블로그 글에서 잠깐 언급한 적 있는데, 아래의 그림과 비슷하게 CL은 빨간선 실선 & UCL, LCL은 파란색 점선으로 시각화하였다.
  - 블로그 글 : 제조업에서 데이터 분석이란?

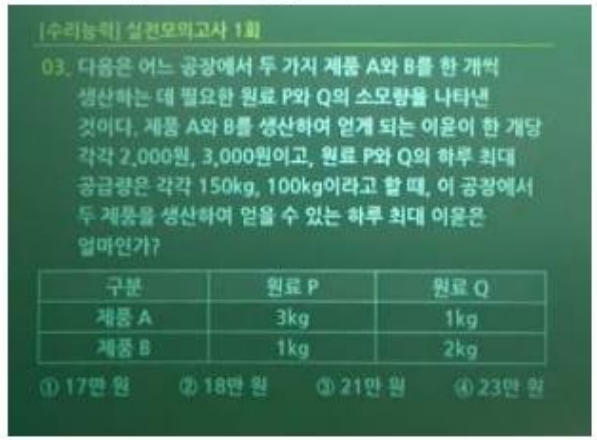



(사진 출처: control chart)

# 2.통계분석(50점)

**3. 데이터 없음 - 표에 제품 1, 2를 만드는데 재료 a, b, c가 일부 사용되며, 제품 1과 2를 만들 때 12만원과 18만원을 벌 수 있다. 재료는 한정적으로 주어지는데, 이때 최대 수익을 낼 수 있을 때의 제품 1과 제품2의 개수를 알고 싶음 (제품 수량을 최대로 뽑으면서 수익이 최적이 되도록)**

- 문제 예시는 아래와 같습니다. 여기서 조금 다른 점은 원료가 3개이고, 제품 1의 경우 하나를 생산할 때 **재료 a를 n개** 얻고 재료 b를 n개 쓰고 재료 c를 n개 쓴다는 것입니다.


[수리능력] 실전모의고사 1회

03. 다음은 어느 공장에서 두 가지 제품 A와 B를 한 개씩 생산하는 데 필요한 원료 P와 Q의 소모량을 나타낸 것이다. 제품 A와 B를 생산하여 얻게 되는 이윤이 한 개당 각각 2,000원, 3,000원이고, 원료 P와 Q의 하루 최대 공급량은 각각 150kg, 100kg이라고 할 때, 이 공장에서 두 제품을 생산하여 얻을 수 있는 하루 최대 이윤은 얼마인가?

| 구분 | 원료 P | 원료 Q |
|---|---|---|
| 제품 A | 3kg | 1kg |
| 제품 B | 1kg | 2kg |

① 17만 원 ② 18만 원 ③ 21만 원 ④ 23만 원

(사진 출처: NCS수리 문제풀이)

- 위의 내용을 방정식으로 바꿔서 풀어내는 문제이며, 연립방정식 문제라고 볼 수 있습니다.
  - 솔직히 전형적인 ncs 문제가 갑자기 왜 나왔는지 이해가 되지 않습니다. 일단 풀라고 해서 풀긴 했는데, 어이없어하면서 for문을 짰던 기억이 납니다.

- 이중 for문을 짰으며 수익값이 최대가 되는 부분과 이때의 제품 1과 제품2의 개수를 print하도록 하였습니다.

# 2.통계분석(50점)

데싸라면 풀이
연관성분석의 기초가 되는 runtest

**4 데이터 없음 - 상품 a와 b가 있을 때 다음과 같은 구매 패턴이 있다고 한다. aa bb aaaa bbbb a b aa bb aa bbb aa bb a b (정확히 기억 안나지만 대충 비슷함) 구매하는 패턴으로 보아 두 상품이 연관이 있는지 궁금함**

- 일단 문제를 보고 욕부터 먼저 했습니다. 비모수 검정인 것 같은데, 데이터 없이 이렇게 주고 검정하라고 할 줄은 몰랐습니다. 어떻게 해야할지 몰라서 손도 못대고 이 문제는 포기하고 넘어갔습니다.
- 다른분들 후기를 보니 **Run Test**를 진행하면 된다고 합니다.
  - Run-test : 일련의 연속적인 관측값들이 임의적(random)으로 나타난 것인지를 검정하는 방법이며, 관측값들이 얻어진 순서에 근거하는 비모수적 검정법
  - 블로그 설명 : Run-test

#### 4.1 연구가설과 귀무가설 작성

- $H_0H_0$ : 연속적인 관측값이 임의적이다. (=표본이 무작위로 추출되었다)

#### 4.2 평균과 표준편차

$$\mu = \frac{2n_1n_2}{n_1+n_2}+1 \qquad \sigma^2 = \frac{2n_1n_2(2n_1n_2-n_1-n_2)}{(n_1+n_2)^2(n_1+n_2-1)} \qquad z = \frac{u-\mu}{\sigma}$$

#### 4.1.3 가설 채택

- tseries 패키지의 `runs.test` 함수를 사용하면 됩니다. (수열은 두가지 수준으로 된 요인어야함)
- 예시코드

```
> s<-sample(c(0,1),100,replace=T)
> runs.test(as.factor(s))

        Runs Test

data:  as.factor(s)
Standard Normal = -1.2346, p-value = 0.217
alternative hypothesis: two.sided
```

# End