데이터마이닝 프로젝트

레포트

신체에 따른 의류사이즈

데이터 마이닝

컴퓨터정보공학부 201520747 심아윤

컴퓨터정보공학부 201621167 이지은

목차

1. 프로젝트 소개 ----------3

2. 도메인 소개----------4

3. Weka----------5

4. 학습곡선----------9

5. ANOVA - 유의성 검사----------10

6. Bernoulli distribution————————12

7. DEMO————————————13

8. 논의 ————————————17

9. 팀별 역할 ————————————18

10. 참고 문헌————————————19

1. 프로젝트 소개

현대인들은 오프라인으로 옷을 구매하기 보다, 온라인 상점을 이용하여 옷을 구매하는 일이 많아졌습니다.

온라인 상점에서도 각자 그들의 기준으로 옷을 만들고, 그에 맞는 사이즈를 제공합니다.

구매자는 쇼핑몰이 제공하는 데이터를 참고하여, 자신에게 알맞은 사이즈를 골라 구매를 합니다.

하지만 사람마다 체형이 다르고, 몸무게와 키가 다름에도 온라인 쇼핑몰에서는 정형화되어진 사이즈를 제공하는 곳이 많이 있습니다.

그렇기 때문에 잘못된 사이즈를 골라 옷을 구매하게 되는 경우가 적지 않게 발생되며,

그러한 실수를 만들지 않기 위해 옷을 구매하기 전 사람들은 자신보다 먼저 구매를 한 사람들의 후기를 통해 옷의 사이즈에 대한 정보를 더 많이 얻고 있습니다.

즉, 정형화되어진 사이즈를 제공하는 온라인 쇼핑몰의 관계자보다 직접 옷을 구매하여 착용한 사용자의 신체 정보를 제공받고, 제공받은 정보를 통해 자신의 사이즈를 예측하여 온라인 구매를 하는 경우가 많아지고 있습니다.

따라서, 저희는 이러한 경우를 반영하여 1000 명의 사람들의 사이즈 데이터 set 을 활용하여 사이즈를 예측하고 사용자에게 알맞은 사이즈를 제공할 수 있는 사이트를 만들기 위해 이 프로젝트를 진행하게 되었습니다.

2. 도메인 소개

나이	TWENTY, THIRTY, FOUTY, FIFTY, SIXTY
키	140~180(BEG, MID, END)
몸무게	40~100(BEG, MID, END)
사이즈	XS, X, M, L, XL, XXL

나이는 10 단위로 설정하였고, 키와 몸무게는 모두 0~3 까지는 BEG, 4~6 까지는 MID, 7~9.99 까지는 END 로 설정하여 인스턴스의 범위를 나누어 진행하였습니다.

	A	B	C	D
1	weight	age	height	size
2	100_BEG	twenty	160_END	XXL
3	100_BEG	thirty	150_MID	XXL
4	100_BEG	thirty	170_MID	XXL
5	40_END	twenty	150_MID	XS
6	40_END	thirty	150_BEG	XS
7	40_END	fourty	150_BEG	XS
8	40_END	thirty	170_BEG	S
9	40_END	twenty	170_BEG	S
10	40_END	thirty	170_MID	S
11	40_END	thirty	160_END	S
12	40_END	fourty	160_END	S
13	40_END	twenty	160_BEG	S
14	40_END	thirty	150_END	S
15	40_END	fourty	150_END	S
16	40_END	twenty	160_END	S
17	40_MID	twenty	150_BEG	XS
18	40_MID	thirty	150_BEG	XS
19	40_MID	thirty	160_BEG	S
20	50_BEG	thirty	160_BEG	XS
21	50_BEG	thirty	150_MID	XS
22	50_BEG	thirty	170_BEG	S
23	50_BEG	twenty	150_END	XS
24	50_BEG	twenty	150_END	XS

3. weka

300 개의 인스턴스를 one-Rule 을 통하여 학습한 결과입니다.

```
weight:
        100_BEG -> XXL
        40_END  -> S
        40_MID  -> XS
        50_BEG  -> XS
        50_END  -> M
        50_MID  -> S
        60_BEG  -> M
        60_END  -> XL
        60_MID  -> L
        70_BEG  -> XL
        70_MID  -> L
        77_END  -> XXL
        80_BEG  -> XXL
        80_END  -> XXL
        80_MID  -> XXL
        90_BEG  -> XXL
        90_END  -> XXL
        90_MID  -> XXL
(187/300 instances correct)


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         182               60.6667 %
Incorrectly Classified Instances       118               39.3333 %
Kappa statistic                          0.5162
Mean absolute error                      0.1311
Root mean squared error                  0.3621
Relative absolute error                 48.3477 %
Root relative squared error             98.3513 %
Total Number of Instances              300

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.702    0.040    0.767      0.702    0.733      0.687    0.831     0.586     XXL
                 0.818    0.040    0.621      0.818    0.706      0.687    0.889     0.521     XS
                 0.737    0.074    0.700      0.737    0.718      0.650    0.831     0.566     S
                 0.718    0.231    0.490      0.718    0.583      0.435    0.743     0.419     M
                 0.338    0.047    0.676      0.338    0.451      0.384    0.645     0.379     L
                 0.429    0.057    0.500      0.429    0.462      0.398    0.686     0.281     XL
Weighted Avg.    0.607    0.095    0.626      0.607    0.597      0.518    0.756     0.455

=== Confusion Matrix ===

  a  b  c  d  e  f   <-- classified as
 33  0  0  3  3  8 |  a = XXL
  0 18  3  1  0  0 |  b = XS
  0  5 42 10  0  0 |  c = S
  0  5 14 51  1  0 |  d = M
  5  1  1 31 23  7 |  e = L
  5  0  0  8  7 15 |  f = XL
```

300 개의 인스턴스를 naive-bayes 을 통하여 학습한 결과입니다.

```
height
  160_END         10.0    4.0   14.0    4.0   14.0    5.0
  150_MID          2.0    4.0    4.0    4.0    2.0    3.0
  170_MID          6.0    1.0    3.0    7.0    3.0    9.0
  150_BEG          1.0    5.0    3.0    2.0    2.0    1.0
  170_BEG         12.0    2.0    4.0   27.0   11.0   11.0
  160_BEG         11.0    8.0   16.0   16.0   30.0    3.0
  150_END          1.0    5.0    8.0   12.0    6.0    2.0
  160_MID          6.0    1.0   12.0    2.0    7.0    3.0
  180_BEG          4.0    1.0    2.0    1.0    1.0    4.0
  170_END          3.0    1.0    1.0    6.0    1.0    4.0
  140_END          1.0    1.0    1.0    1.0    2.0    1.0
  180_MID          2.0    1.0    1.0    1.0    1.0    1.0
  [total]         59.0   34.0   69.0   83.0   80.0   47.0


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         208               69.3333 %
Incorrectly Classified Instances        92               30.6667 %
Kappa statistic                          0.6206
Mean absolute error                      0.1785
Root mean squared error                  0.2832
Relative absolute error                 65.8245 %
Root relative squared error             76.9339 %
Total Number of Instances              300

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.660    0.036    0.775      0.660    0.713      0.667    0.907     0.771     XXL
                 0.818    0.029    0.692      0.818    0.750      0.731    0.945     0.811     XS
                 0.860    0.078    0.721      0.860    0.784      0.732    0.938     0.804     S
                 0.732    0.118    0.658      0.732    0.693      0.593    0.875     0.667     M
                 0.662    0.103    0.652      0.662    0.657      0.555    0.886     0.687     L
                 0.371    0.019    0.722      0.371    0.491      0.477    0.846     0.590     XL
Weighted Avg.    0.693    0.076    0.697      0.693    0.686      0.619    0.896     0.716


=== Confusion Matrix ===

  a  b  c  d  e  f   <-- classified as
 31  1  1  4  8  2 |  a = XXL
  0 18  3  1  0  0 |  b = XS
  0  3 49  5  0  0 |  c = S
  0  3 10 52  6  0 |  d = M
  4  1  3 12 45  3 |  e = L
  5  0  2  5 10 13 |  f = XL
```

400 개의 인스턴스를 J48 을 통하여 학습한 결과입니다.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         281               70.3333 %
Incorrectly Classified Instances       119               29.6667 %
Kappa statistic                          0.6136
Mean absolute error                      0.1287
Root mean squared error                  0.2603
Relative absolute error                 48.7456 %
Root relative squared error             71.6674 %
Total Number of Instances              400

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.797    0.051    0.764      0.797    0.780      0.733    0.941     0.858     XXL
                 0.441    0.029    0.722      0.441    0.547      0.510    0.906     0.686     S
                 0.731    0.003    0.950      0.731    0.826      0.824    0.928     0.843     XS
                 0.866    0.238    0.629      0.866    0.728      0.589    0.870     0.733     M
                 0.682    0.044    0.806      0.682    0.739      0.679    0.887     0.737     L
                 0.382    0.033    0.520      0.382    0.441      0.403    0.873     0.337     XL
Weighted Avg.    0.703    0.101    0.715      0.703    0.695      0.621    0.895     0.722

=== Confusion Matrix ===

   a   b   c   d   e   f   <-- classified as
  55   0   0   4   4   6 |   a = XXL
   0  26   1  32   0   0 |   b = S
   0   5  19   2   0   0 |   c = XS
   2   5   0 110   6   4 |   d = M
   4   0   0  21  58   2 |   e = L
  11   0   0   6   4  13 |   f = XL
```

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     400개-표 1
Instances:    400
Attributes:   4
              weight
              age
              height
              size
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

weight = 100_BEG: XXL (3.0)
weight = 40_BEG: S (1.0)
weight = 40_END
|   height = 160_END: S (3.0)
|   height = 150_MID: XS (1.0)
|   height = 170_MID: M (1.0)
|   height = 150_END: XS (2.0)
|   height = 150_BEG: XS (2.0)
|   height = 170_BEG: M (2.0)
|   height = 160_BEG: S (2.0)
|   height = 160_MID: S (0.0)
|   height = 180_BEG: S (0.0)
|   height = 170_END: S (0.0)
|   height = 140_END: S (0.0)
|   height = 180_MID: S (0.0)
weight = 40_MID: XS (4.0/1.0)
weight = 50_BEG
|   height = 160_END: S (7.0)
|   height = 150_MID: XS (5.0)
|   height = 170_MID: XS (1.0)
|   height = 150_END: XS (7.0)
|   height = 150_BEG: XS (2.0)
|   height = 170_BEG: M (4.0/2.0)
|   height = 160_BEG: S (15.0/3.0)
|   height = 160_MID: S (2.0)
|   height = 180_BEG: S (0.0)
|   height = 170_END: S (0.0)
|   height = 140_END: S (0.0)
|   height = 180_MID: S (0.0)
weight = 50_END: M (62.0/27.0)
weight = 50_MID: M (66.0/31.0)
weight = 60_BEG
weight = 60_END: XL (22.0/9.0)
weight = 60_MID: L (43.0/13.0)
weight = 70_BEG: XXL (29.0/14.0)
weight = 70_MID
|   height = 160_END: XXL (1.0)
|   height = 150_MID: XXL (0.0)
|   height = 170_MID: XXL (0.0)
|   height = 150_END: L (2.0)
|   height = 150_BEG: XXL (0.0)
|   height = 170_BEG: XXL (2.0/1.0)
|   height = 160_BEG: XXL (0.0)
|   height = 160_MID: XXL (0.0)
|   height = 180_BEG: XXL (0.0)
|   height = 170_END: XL (1.0)
|   height = 140_END: XXL (0.0)
|   height = 180_MID: XXL (0.0)
weight = 70_END: XXL (1.0)
weight = 77_END: XXL (18.0)
weight = 80_BEG: XXL (11.0)
weight = 80_END: XXL (1.0)
weight = 80_MID: XXL (2.0)
weight = 90_BEG: XXL (3.0)
weight = 90_END: XXL (1.0)
weight = 90_MID: XXL (3.0)

Number of Leaves  :     64

Size of the tree :      69
```

4. 학습곡선

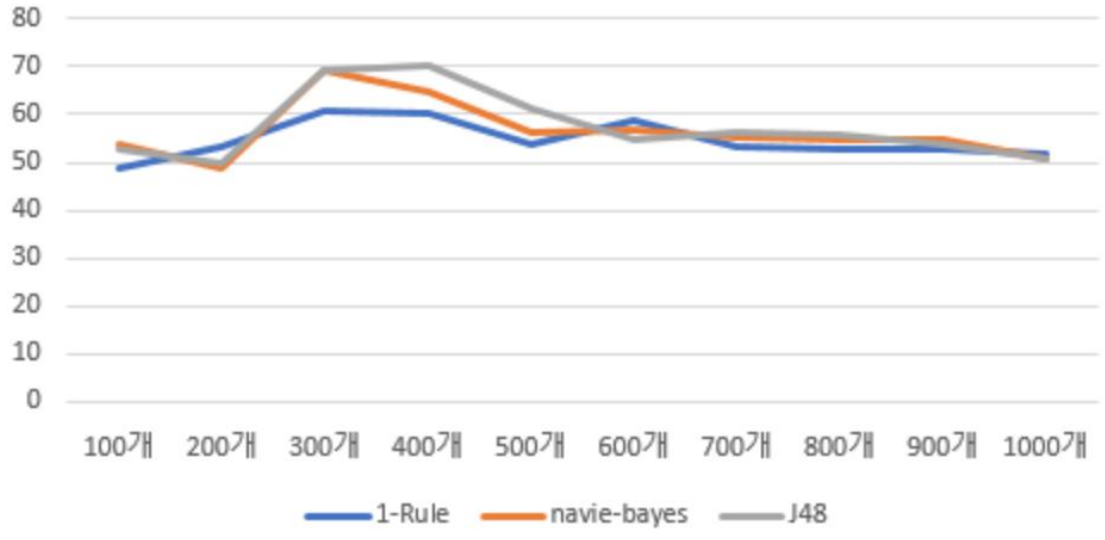
아래의 표는 저희가 이용한 알고리즘 OneRule, J48, Naïve-Bayes 를 이용하여 얻은 정확도입니다.

	100개	200개	300개	400개	500개	600개	700개	800개	900개	1000개
1-Rule	30	43.5	60.6667	60	54	58.5	53.1429	52.875	52.6667	49.8498
navie-bay	33	49	69.3333	64.75	56.4	56.8333	55.2857	54.875	55	49.7
J48	32	50	69	70.3333	61	55	56.1429	56	53.7778	50.6507

총 1000 개의 인스턴스를 사용하였으며, 표의 왼쪽부터 순서대로 100 개부터 10%씩 증가한 인스턴스들의 각 정확도를 표로 나타낸 것입니다.

아래의 표를 보시면 J48 알고리즘을 제외한 나머지 OneRule, Naïve-Bayes Rule 에서는 300 개의 인스턴스를 가질 때에 각각 OneRule : 60.6667, Naïve-Bayes Rule : 69.3333 의 가장 높은 정확도를 얻을 수 있었고, J48 알고리즘에서는 인스턴스의 수가 400 일 경우 70.3333 의 3 개의 알고리즘을 통틀어 가장 높은 정확도를 얻을 수 있었습니다.

위의 표를 이용하여 얻은 그래프는 아래의 그래프와 같으며, OneRule 과 Naïve-Bayes Rule 은 300 개의 데이터에서 가장 높은 정확도를 보였고, J48 은 400 개의 데이터에서 가장 높은 정확도를 얻을 수 있음을 표보다 쉽게 한눈에 알 수 있습니다.



5. ANOVA - 유의성 검사

위의 학습 곡선에서 최고치인 300 개의 인스턴스를 기준으로 하여, naive-bayes, oen-Rule 알고리즘에서 각각 280 개, 290 개, 300 개, 310 개, 320 개의 인스턴스, J48 알고리즘에 대해서는 380 개, 390 개, 400 개, 410 개, 420 를 통해 얻은 정확도를 아래의 표로 나타냈습니다.

인스턴스 수	OneRule	J48	인스턴스 수	Navie-Bayes
280	58.125	65.625	380	66.25
290	59.6774	67.4194	390	70.3448
300	60	70.3333	400	69.333
310	58.9286	68.2759	410	67.4194
320	60.3448	70.3571	420	67.8571

위의 표를 엑셀의 데이터 분석의 분산 분석 - 일원 배치법에 적용하여 아래의 표와 같이 합, 평균, 분산을 얻고, 분산 분석의 결과를 얻었습니다.

분산 분석: 일원 배치법

요약표

인자의 수준	관측수	합	평균	분산
Column 1	5	297.0758	59.41516	0.794073
Column 2	5	342.0107	68.40214	4.061367
Column 3	5	341.2043	68.24086	2.601231

분산 분석

변동의 요인	제곱합	자유도	제곱 평균	F 비	P-값	F 기각치
처리	264.4747	2	132.2373	53.20229	1.08E-06	3.885294
잔차	29.82669	12	2.485557			
계	294.3014	14				

정확도의 기각치의 기준은 3.885294 이며, 저희의 데이터의 정확도의 비는 53.20229 를 얻음으로써 우연성이 기각되기에 충분한 조건이 됨으로 통계적 유의성이 있다고 판단할 수 있는 데이터임을 보였습니다.

저희는 ANOVA 유의성을 검사하기 전, 온라인 쇼핑몰에서 후기에 사용자의 신발사이즈 정보를 함께 제공하기 때문에 그 부분을 반영하고자 신발사이즈 데이터를 함께 만들어 프로젝트를 진행하였습니다.

ANOVA 유의성을 검사하기 전도 역시 Weka 를 통해 얻는 정확도가 낮은 정확도 값을 보였고, 아래의 표와 같이 신발사이즈에 대한 ANOVA 유의성을 검사를 진행한 결과 데이터의 분산 분석 결과 역시 정확도의 기각치를 넘기지 못하였습니다.

따라서 신발사이즈 데이터는 통계적 유의성에 불충분한 데이터로 증명이 되었고, 프로젝트에서는 제외된 데이터가 되었습니다.

분산 분석: 일원 배치법

요약표

인자의 수준	관측수	합	평균	분산
Column 1	10	376.9175	37.69175	53.55774
Column 2	10	453.9722	45.39722	178.3959
Column 3	10	424.9513	42.49513	22.27555

분산 분석

변동의 요인	제곱합	자유도	제곱 평균	F 비	P-값	F 기각치
처리	302.8962	2	151.4481	1.787145	0.186681	3.354131
잔차	2288.063	27	84.74306			
계	2590.959	29				

6. Bernoulli distribution

베르누이 계산을 각 알고리즘에 대해 진행하였습니다.

F 의 값은 각 알고리즘의 최대의 정확도를 나타낼 때의 정확도를 이용하였으며, 최고의 정확치를 나타낼 때의 인스턴스 수인 400 을 N 으로 하였습니다.

각 알고리즘의 베르누이 분산 결과를 아래의 표로 정리하여 나타냈습니다.

<OneR> (F = 60%, N = 300)

C = 80%	C = 90%	C = 95%	C = 98%	C = 99%
P[0.563,0.636]	P[0.553,0.646]	P[0.544,0.654]	P[0.533,0.663]	P[0.526,0.670]
56.3%<P<63.6%	55.3%<P<64.6%	54.4%<P<65.4%	53.3%<P<66.3%	52.6%<P<67.0%

<J48> (F = 70%, N = 400)

C = 80%	C = 90%	C = 95%	C = 98%	C = 99%
P[0.670,0.729]	P[0.661,0.736]	P[0.653,0.743]	P[0.644,0.750]	P[0.638,0.755]
67.0%<P<72.9%	66.1%<P<73.6%	65.3%<P<74.3%	64.4%<P<75.0%	63.8%<P<75.5%

<Naïve-Bayes> (F = 69%, N = 300)

C = 80%	C = 90%	C = 95%	C = 98%	C = 99%
P[0.655,0.723]	P[0.645,0.732]	P[0.636,0.740]	P[0.625,0.749]	P[0.618,0.754]
65.5%<P<72.3%	64.5%<P<73.2%	63.6%<P<74.0%	62.5%<P<74.9%	61.8%<P<75.4%

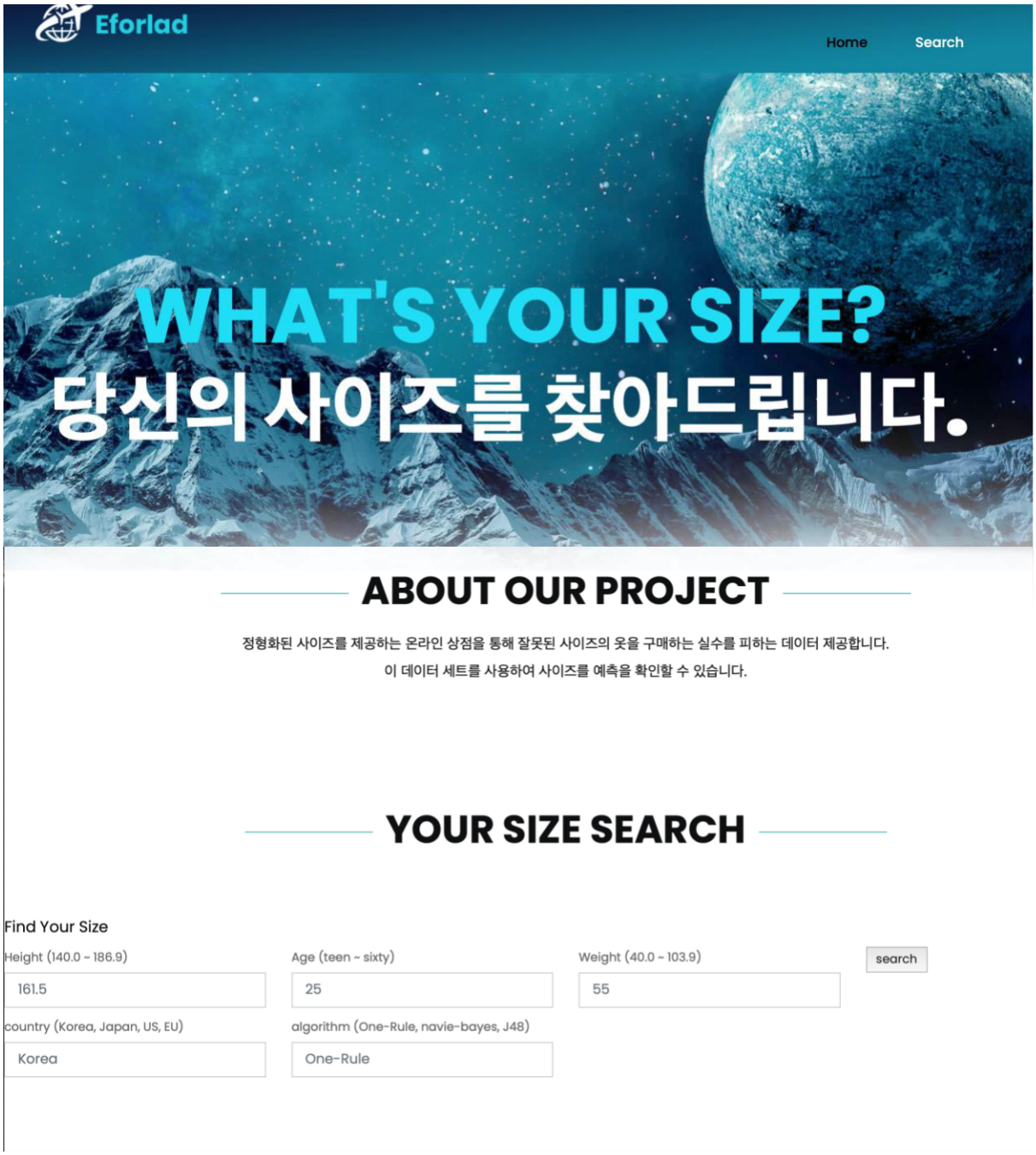
각 알고리즘의 베르누이 분산 측정의 공통된 방식은 일정한 값의 F(정확도)와 N(인스턴스 수)이 주어질 때 C 의 신뢰도 퍼센트만을 80%에서 99%까지 변화시켜 5 번에 거쳐 진행하였습니다.

이 때, 위의 각 표에서도 확인할 수 있듯이 F 와 N 이 일정하고, C 의 신뢰도 퍼센트가 높아질수록 P 의 범위가 넓어짐을 확인할 수 있습니다.

즉, C 인 신뢰도가 커질수록 P 인 성공률의 범위는 점차 넓어짐을 확인할 수 있습니다.

7. DEMO

저희의 처음 실행화면입니다.

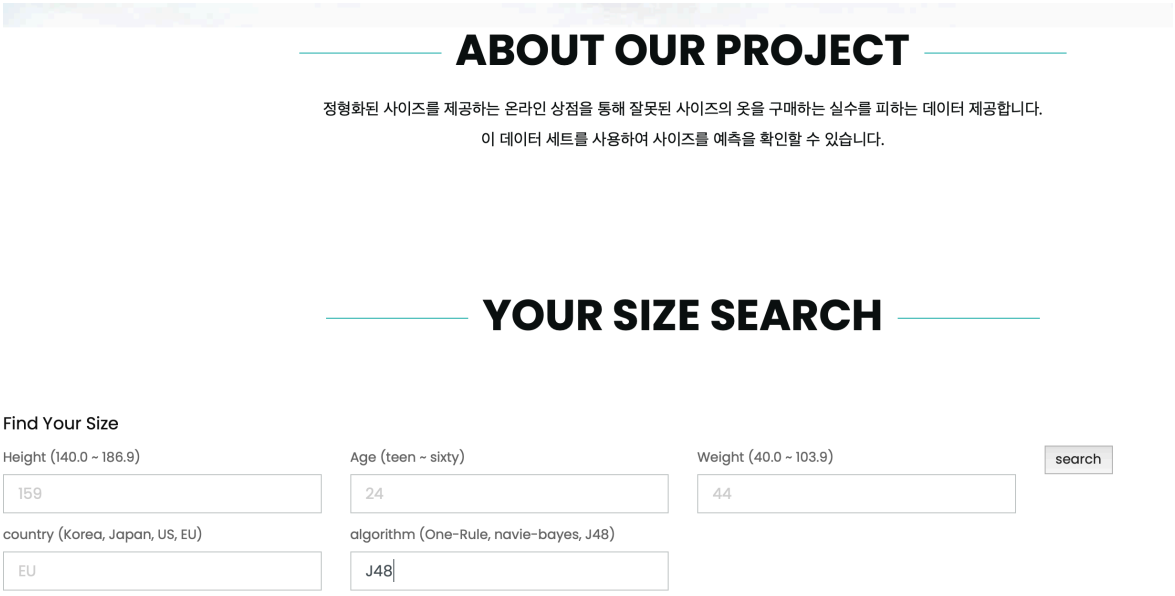


의류사이즈 예측 웹사이트를 제작하였고, 추가적으로 각 나라별 사이즈로 매핑이 될 수 있도록 구매하고자 하는 의류 제조국을 기입하는 칸을 추가하였습니다.

각 테스트 데이터에 따른 결과 화면 창입니다.

Testcase1 : 나이: 24, 키:159, 몸무게:44, 제조국: EU, 알고리즘: J48 의 결과화면입니다.

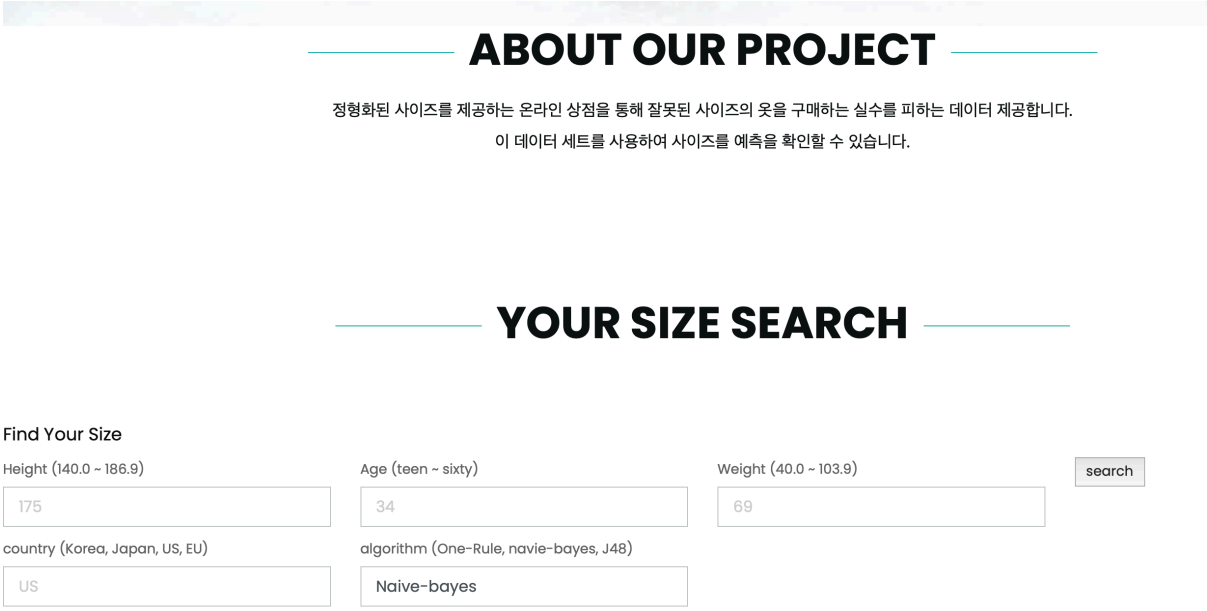
[입력]



[결과]

Testcase2 : 나이: 34, 키:175, 몸무게:69, 제조국: US, 알고리즘: naive-bayes 의 결과화면입니다.

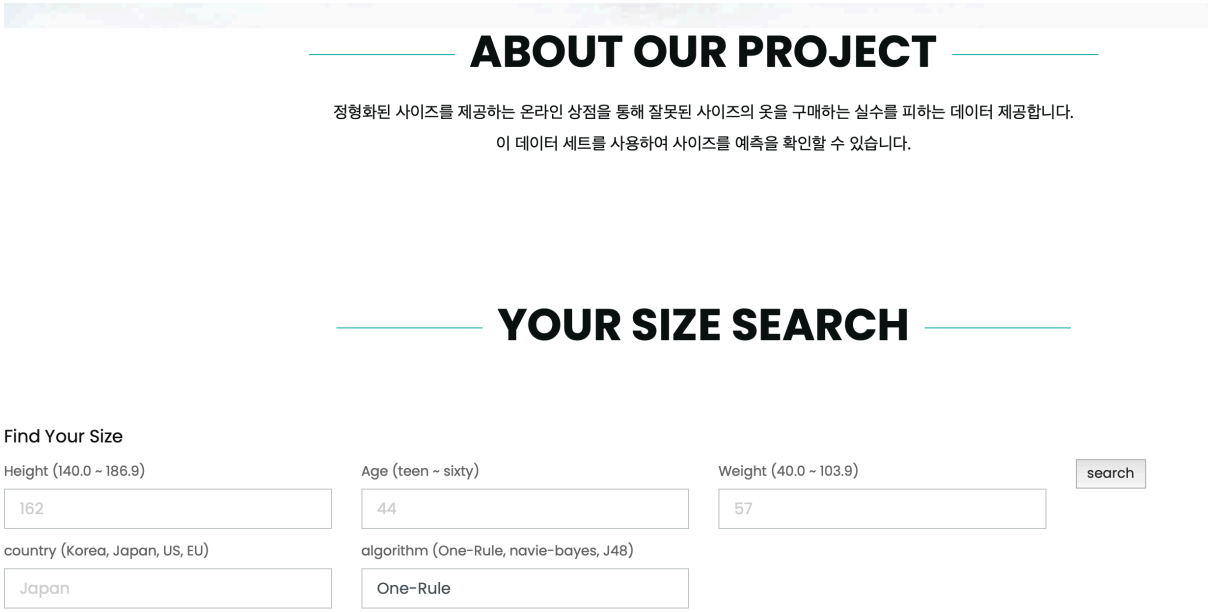
[입력]



[결과]

Testcase3 : 나이: 44, 키:162, 몸무게:57, 제조국: Japan, 알고리즘: One-Rule 의 실행결과화면입니다.

[입력]



[결과]

위와 같이 사이즈를 탐색한 결과(S, M, L)와 기본적으로 한국사이즈(44, 55, 66, 77, 88) 에 매핑되는 사이즈와 자신이 선택한 나라의 사이즈로 매핑된 결과, 정확도를 출력하게 하였습니다

8. 논의

1-Rule, Naive-bayes, J48 각 알고리즘에 대해서 최고 정확도는 각각 60.6667%, 69.3333%, 70.3333% 라는 결과값을 얻었습니다. 또한 이 프로젝트에서는 사이즈를 예측하기 위해서 키, 몸무게, 나이 3가지의 속성을 기준으로 클래스 값을 도출하도록 하였습니다. 하지만 정확한 사이즈 예측을 위해서는 이와 같은 속성 뿐만 아니라, 사이즈를 선택하는 개인의 취향, 개인별 체형 데이터가 필요하지만 이 같은 데이터가 반영되어 있지 않았습니다. 저희는 이러한 더 많은 속성을 반영한 데이터를 활용했다면, 더 높은 정확도를 얻을 수 있을 것이라고 추측합니다.

베르누이 과정을 진행하면서 얻은 신뢰도에 따른 각 알고리즘 별, 5개의 정확도에 대한 분포를 분석해보면 , 같은 인스턴스에 대해서 신뢰도가 높아질 수록 정규분포에 따라, Zα 값이 커지고 이에 해당되는 Φ(Zα)-Φ(-Zα)의 값 역시 커짐을 알 수 있습니다. 이러한 사실을 각 신뢰도가 80%일 경우부터, 99% 일 때까지의 측정한 정확도 값의 범위가 점차 커진다는 결과를 통해 확인할 수 있었습니다.

의류사이즈 예측 프로젝트에서 활용한 데이터 셋은 성별이 여자인 경우에만 한정적으로 데이터를 분석하였습니다. 고로, 결과적으로 구현한 demo 프로그램 또한 여성 의류 사이즈 예측에 한정되어 있습니다. 이 프로그램이 남성 여성 구분없이 통용되기 위해서는 여성뿐만 아니라 남성 데이터 또한 고려해야합니다.

저희는 의류사이즈 예측 프로젝트와 함께 신발사이즈 예측을 추가적으로 진행했습니다. 하지만 이 ANOVA 유의성 검사에서 F비가 F기각치를 넘기지 못해 귀무가설을 기각하지 못하였습니다. 신발데이터는 신발사이즈(클래스)의 수는 매우 많지만 그에 비해 인스턴스의 수가 약 총 100개였습니다. 이러한 측면에서 충분한 데이터를 확보하지 못한 점이 데이터의 통계적 규칙이나 패턴을 알아내기 어려웠고, 결론적으로 통계적으로 유의미한 데이터가 아니라는 사실을 얻게 되었다고 생각합니다.

9. 팀별 역할

심아윤 : 데이터 수집, 자료 분석, 코드 구현, 레포트 작성, 발표

이지은 : 데이터 수집, 자료 분석, 코드 구현, 레포트 작성, PPT

10. 참고 문헌

https://www.kaggle.com/datasets

https://weka.sourceforge.io/doc.dev/weka/classifiers/package-summary.html