

심혈관 질환 진단 엔진의 개발과 시각화

161138 조성현

목차

I. 서론

1. 연구 배경 및 필요성
2. 문헌 조사
3. 연구 목적, 방향, 예상 결과

II. 본론

1. 문제 정의 (Problem Definition)
2. 해결책 설계 (Solution Planning)
3. 실행 (Execution)
4. 검증 (Testing)
5. 배포 (Distribution)

III. 결론

1. 결론 및 기대 효과
2. 문제점 및 향후 보완 계획

IV. 부록

1. 참고 자료
2. 소스 코드 (Source Code)
3. 의사 결정 트리 시각화 이미지
4. 학습에 활용된 데이터 세트 전체

I. 서론

연구 배경 및 필요성

지난 10년 동안, 심장 질환으로 인한 사망률은 약 40% 증가했다. 조만간 우리나라 사망률 1위에 오를지도 모른다는 전문가들의 염려에도 심장 질환을 조기에 발견하지 못하여 사망하는 사람들이 있고, 이 중에는 의료 사각지대에 놓인 사람이 많다. 만약 간단한 인공지능을 이용해서 심장 질환의 가능성을 알아볼 수 있다면 부정확하더라도 큰 의미를 가질 수 있을 것이다.

문헌 조사

해당 분야는 연구가 상당히 진척되어있다. 이 중 가장 지도적인 역할은 Apple으로, 2018년 Stanford University와 제휴하여 미국 내에서 Apple Watch 이용자를 대상으로 심장 연구를 진행한 바 있다. Apple은 해당 기능에 대한 연구를 완료하고 자사 제품의 심박 건강 확인 기능에 도입하였다. 인공지능 라이브러리 개발의 선두를 달리고 있는 Google 또한 환자의 눈을 보고 심장 질환을 진단할 수 있는 알고리즘 개발에 착수한 바 있다. 이처럼 심혈관 질환은 사람의 생명에 치명적인 질환이기 때문에, 많은 기업들이 선도적으로 연구를 진행하고 있다.

반면 개인이 이런 연구를 진행한 선례는 많지 않다. 이런 연구는 시간과 자원이 많이 들어가기 때문이다. 전문적인 인공지능의 개발 또한 매우 어렵지만, 데이터 자체를 구하는 것이 가장 어렵다. 안정적인 인공지능을 개발하기 위해서는 작게는 수 천, 많게는 수십 억 개의 데이터가 기본적으로 필요하다. 심장 질환의 연구 같은 경우에는 환자 혹은 일반인들이 각각 데이터가 하나가 되므로, 최소한 수 백에서 수 천 명의 환자들의 데이터를 수집해야 한다는 의미가 된다. 개인이 이런 규모의 연구를 진행하기에는 무리가 있으며, 때문에 공공 기관 혹은 기업이 데이터를 오픈 소스로 공유하면 이를 통해 개인들이 인공지능 모델을 자유롭게 개발하는 경우가 많다.

연구 목적, 방향과 예상 결과

해당 연구 또한 마찬가지로의 방법을 통할 것이다. 궁극적인 연구 목적은 우리나라 국민들을 대상으로 심혈관 질환 데이터를 수집하여 모두가 다운로드하고 사용할 수 있는 인공지능 기반 앱을 개발하는 것이다. 하지만 이는 장기간 데이터를 수집하고 연구 및 개발을 진행해야 하는 사항으로, 본 연구에서는 그 초석을 마련하기 위해 오픈 소스로 공개된 심혈관 질환 정보를 이용하여 심혈관 질환 진단 엔진을 개발하고 해당 엔진의 의사 결정 트리를 시각화해보려 한다.

II. 본론

문제 정의 (Problem Definition)

해당 연구는 University of California Irvine의 Machine Learning Repository의 일환으로 공개한 Heart Disease UCI 데이터 세트(일명 '클리블랜드 데이터 세트')를 이용할 것이다. 원본 데이터는 총 76개의 문항을 가지고 있지만 공개된 데이터 세트는 14개의 문항을 가지고 있다. 총 303개의 익명 데이터가 공개되어 있으며, 데이터는 다음과 같이 제공된다. 데이터 전체가 .csv 확장자로 된 스프레드시트의 형태이며, 부록에 첨부되어 있다.

클리블랜드 데이터 세트

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...

이 중 각 문항이 나타내는 정보를 구체화하자면 다음과 같다. 이 설명의 출처는 부록에 첨부되어 있다.

문항	한글명	설명	유형	타입
age	나이	나이가 숫자로 주어진다.	수치	int
sex	성별	여성은 0, 남성은 1의 범주로 구분되어 주어진다.	범주	int
cp	가슴 통증 유형	가슴 통증 유형을 0에서 4로 나눈다.	범주	int
trestbps	안정 시 혈압	병원 입원 시 혈압이 mmHg로 주어진다.	수치	int
chol	혈청 콜레스테롤	혈청의 콜레스테롤 농도가 mg/dl로 주어진다.	수치	int
fbs	공복 혈당	공복 시 혈당이 120mg/dl를 초과하지 않는다면 0, 초과한다면 1의 값이 주어진다.	범주	int
restecg	안정 심전도 결과	안정 심전도 결과가 0, 1, 2로 구분하여 주어진다.	범주	int
thalach	최대 심박수	최대 심박동수가 수치로 주어진다.	수치	int
exang	협심증 유발 양상	심장 운동이 협심증을 유발하지 않는다면 0, 유발한다면 1의 값이 주어진다.	범주	int
oldpeak	운동 유발 ST 하강	운동으로 유발되는 ST 분절 하강 (심장 박동 직후 심장의 운동)의 값이 수치로 주어진다.	수치	int
slope	ST 분절 기울기	ST 분절의 최대 기울기가 수치로 주어진다.	수치	int
ca	주요 혈관 개수	형광 투시된 주요 혈관의 수가 0에서 3의 값으로 주어진다.	수치	int
thal	심장 결함 유형	심장 결함의 유형이 없음, 해결된 결함, 해결 가능한 결함으로 구분되어 주어진다.	범주	String
target	심장병 진단	심장병이 발병하였으면 1, 발병하지 않았으면 0의 값을 가진다.	범주	int

해결책 설계 (Solution Planning)

다음의 과정을 통해 기계 학습을 시킬 것이다. 또한 데이터를 가공하기 위해 다음의 외부 Library를 사용할 예정이다. 전부 Python 3에 기반해 있다.

1. 데이터의 입력

데이터를 컴퓨터에 입력하는 과정이다. 이 과정에서 Pandas를 사용하며, 수학적 데이터를 Python에서 다룰 수 있도록 도와주는 라이브러리이다. Pandas를 활용하여 원본 .csv 파일을 읽어와서 Pandas 데이터 프레임으로 변환하여 사용한다.

2. 데이터의 전처리 (Preprocessing)

기계 학습을 할 수 있도록 데이터를 전처리하는 과정이다. 이 과정에서 Scikit-Learn (sklearn) 라이브러리를 활용하며, Python에서 간단한 기계 학습을 도와주는 라이브러리이다. 전처리는 다음의 과정을 거친다.

1. 입력 데이터를 기계 학습이 가능한 형태로 변환한다.

(`sklearn.preprocessing.LabelEncoder` 사용)

2. 변환된 데이터를 학습 데이터와 검증 데이터로 분류한다. 이때 데이터는 절반으로 나눈다. 즉, 303개의 데이터 중 151개는 훈련용이며, 152개는 검증용으로 사용된다.

(`sklearn.model_selection.train_test_split` 사용)

3. 데이터의 학습

데이터는 패턴 인식과 분류(Grouping) 혹은 의사 결정 트리 제작, 총 2가지 방법으로 이용할 것이다. 이 과정에서 Support Vector Machine과 Tree를 활용한다.

1. 훈련 데이터에서 독립 변수 13개와 종속 변수 1개를 분리하여 각각 Xtrain, Ytrain으로 둔다.
2. SVM과 Tree를 활용해 각각 Input에 Xtrain, Output에 Ytrain을 두고 모델을 학습시킨다.

(`sklearn.svm` 사용)

4. 학습 데이터의 검증

1. 검증 데이터에서 독립 변수 13개와 종속 변수 1개를 분리하여 각각 Xcheck, Ycheck로 둔다.
2. 학습된 모델에 Xcheck를 대입한 뒤 구해지는 Ypredict를 Ycheck로 채점한다. 정답인 경우 1점을 부여하고 오답일 경우 0점을 부여해서 평균 점수를 구한다.

(`sklearn.metrics` 사용)

5. 데이터의 시각화

의사 결정 트리의 경우 데이터를 시각화한다.

(`sklearn.tree`, `graphviz` 사용)

실행 (Execution)

개발은 Google Colab이라는 온라인 개발 환경을 활용했다. Jupyter Notebook이라는 환경을 통해 개발했는데, Jupyter Notebook은 상호 작용이 가능한 여러 코드 블록으로 구성되어 있어 여러 모듈을 동시에 검증할 때 유용하다. 코드의 전문은 부록에 첨부되어 있다.

유의할 점은, 완성된 모델은 총 2가지라는 점이다. 하나는 Support Vector Machine을 활용한 범주화 모델(`sklearn.svm.classes.SVC` object 형태)이고, 나머지 하나는 Tree와 Node를 활용한 의사 결정 트리(`sklearn.tree.tree.DecisionTreeClassifier` Object 형태)의 형태이다. 각각의 경우 모두 `sklearn` 모듈의 함수를 활용했다.

검증 (Testing)

미리 분리해놓은 검증 데이터 세트를 대입하여 값을 계산해보았다. Support Vector Machine을 활용한 경우 0.7894736842105263 점을 획득했다. 즉 약 78.95%의 적중률을 가졌다.

의사 결정 트리의 경우 0.7697368421052632 점을 획득했다. 즉 약 76.97%의 적중률을 가졌다. 또한 의사 결정 트리에서 부록과 같은 이미지를 추출해낼 수 있었다.

배포 (Distribution)

모든 코드와 결과물, 그리고 의사 결정 트리의 벡터 이미지 형태를 GitHub에 업로드하였다. GitHub는 전 세계 수많은 프로그래머들이 자유롭게 자료를 공유하는 오픈 소스 커뮤니티이다.

<https://github.com/anaclumos/HeartDiagnosisEngine>

or



III. 결론

결론 및 기대 효과

본 연구를 통해 SVM와 의사 결정 트리를 이용해 약 75%에서 80%의 정확도를 가지는 학습 모델을 제작할 수 있음을 확인했다. 이와 같은 모델은 질병의 명확한 분별로 인한 장점도 가지지만, 무엇보다 주변 장비들과 복합적으로 연계되기 시작할 때 현실적인 파급력을 가진다. 이와 같은 모델이 90% 이상의 정확도를 가질 때까지 학습된다면, 기존에는 병원에 방문하지 못했던 사람도 자신의 의료 정보를 주변 스마트 기기를 활용해 측정하고 자신의 심장 질환 여부를 예측할 수 있는 것이다. 최근 스마트 기기들이 측정할 수 있는 데이터가 풍부해지고 있기 때문에, 머지않은 미래에 자신의 스마트 기기만으로 학습 모델의 모든 독립 변수를 측정할 수 있는 날이 온다면 전례 없던 큰 규모의 사람들이 자신의 스마트 기기만으로 심혈관 질환을 간편하게 진단할 수 있을 것이다.

문제점 및 향후 보완 계획

본 연구는 여러 가지 한계점을 가진다.

첫째, 데이터 자체에 문제가 있다. 해당 데이터는 1988년에 제작되었다. 30년 전에 비해, 인류는 생활 습관 및 소비 성향, 식사 문화 등이 모두 종합적으로 달라졌다. 때문에 과거의 자료를 가지고 학습을 해도 현대의 인류에 대입하기에는 문제가 있다. 또한 이 자료는 미국인을 기준으로 연구된 데이터이다. 해당 데이터는 서양인을 기준으로 제작되었기 때문에, 우리나라 사람들에게 적용하기에 무리가 있다. 마지막으로 300명은 일반화된 답을 구하기에는 적은 값이다. 다만 이 문제는 데이터 자체의 문제이므로, 데이터만 새롭게 제작할 수 있다면 그 데이터를 다시 대입하고 코드를 재실행하여 새로운 모델을 구하면 된다.

둘째, 심혈관 질환의 가능성은 13개의 항목으로 구하기 어려울 수 있다. 특히 건강과 관련된 질환은 영향을 줄 수 있는 요인이 수도 없이 다양하다. 이런 수많은 요인을 무시하고 13개의 요약된 항목들만으로 모델을 학습하는 것에는 한계가 있을 수밖에 없다. 이 또한 데이터 자체의 문제이긴 하지만, 제작한 모델 또한 13개의 학습 항목을 이용하기 때문에 이 문제에 개선하기 위해서는 데이터의 보강과 모델의 재설계가 같이 필요하다.

셋째, 심혈관 질환의 일반식은 범주화와 의사 결정 트리로 구할 수 없는 형태일지도 모른다. 이를 보완하기 위해 연구자들은 심층 인공신경망 등을 도입하여 해결하고 있다.

궁극적으로 이 연구를 보완하기 위해 필요한 것은 데이터의 항목과 응답자 수 모두에 대한 데이터 보강과, 더 정확한 학습 방식을 사용하는 것이다. 데이터의 보강을 위해서는 병원 혹은 기업 등의 협력이 필요하며, 더 정확한 학습 방식을 구하기 위해서는 고차원적인 학습 패러다임을 배우는 것뿐만이 아니라 심혈관 질환에 대해서도 깊은 이해를 가지고 문제에 접근해야 할 것이다.

IV. 부록

1. 참고 자료

1. Apple 심혈관 질환 연구

Apple의 심혈관 질환 연구에 대한 세부 정보를 참고했습니다.

<https://med.stanford.edu/appleheartstudy.html>

2. Google 인공지능 뉴스

Google의 심혈관 질환 인공지능 연구 동향에 대한 세부 정보를 참고했습니다.

<https://www.theverge.com/2018/2/19/17027902/google-verily-ai-algorithm-eye-scan-heart-disease-cardiovascular-risk>

3. Kaggle

클리블랜드 데이터 세트 설명에 대한 세부 정보를 참고했습니다.

<https://www.kaggle.com/ronitf/heart-disease-uci>

4. Pandas

Pandas 사용법에 대한 세부 정보를 참고했습니다.

<https://pandas.pydata.org/>

5. Scikit Learn

sklearn 사용법에 대한 세부 정보를 참고했습니다.

<https://scikit-learn.org/>

2. 소스 코드 (Source Code)

소스 코드 전문은 실행 가능한 형태로 변환되어 GitHub에 업로드되어 있습니다.

```
# Start of code
import pandas as pd
from sklearn.model_selection import train_test_split as splitter
from sklearn.preprocessing import LabelEncoder as labeler
from sklearn import metrics
from sklearn import svm
from sklearn import tree

Dataframe = pd.read_csv("heart.csv")
Dataframe.columns = ['나이', '성별', '가슴 통증 유형', '안정 혈압 (mmHg)', '혈청 콜레스테롤 (mg/dl)',
                    '공복 혈당 > 120 mg/dl', '안정 심전도 결과 (0, 1, 2)', '최대 심박동수',
                    '협심증 유발 운동', '운동에 의해 유발된 ST Depression',
                    '최대 운동 ST Segment의 기울기', '형광 투시된 주요 혈관의 수 (0-3)',
                    '심장 결함 종류', '심장병 진단']

for column in Dataframe.columns:
    if Dataframe[column].dtype == type(object):
        l = labeler()
        Dataframe[column] = l.fit_transform(Dataframe[column])

Xtrain, Xcheck, Ytrain, Ycheck = splitter(Dataframe.iloc[:, 0:-1], Dataframe.iloc[:, -1],
                                          test_size = 0.5, random_state = 100)

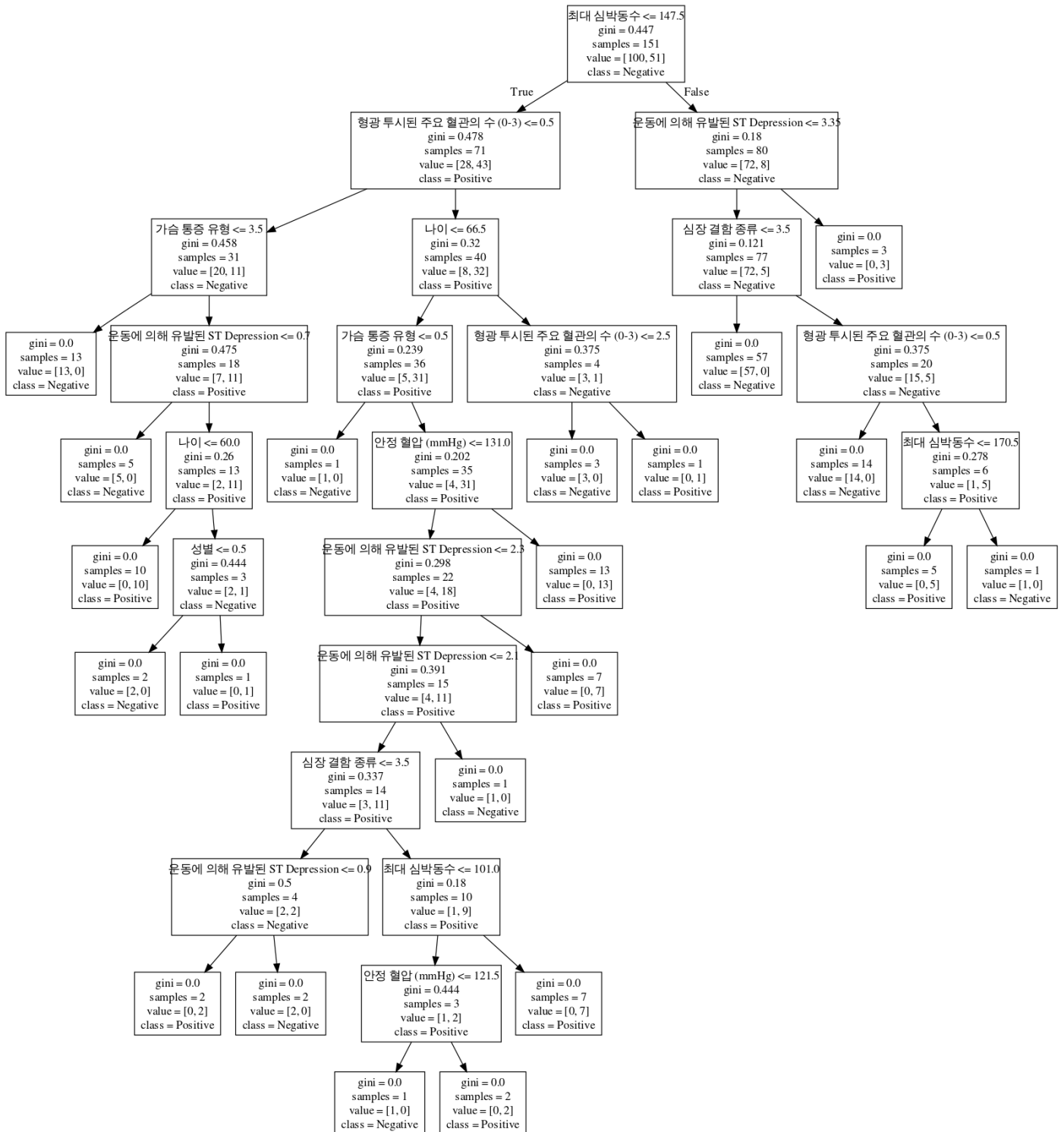
clf = svm.SVC(gamma = 1.0)
clf.fit(Xtrain, Ytrain)
Ypredict = clf.predict(Xcheck)
print("accuracy: ", metrics.accuracy_score(Ycheck, Ypredict))

clf = tree.DecisionTreeClassifier()
clf.fit(Xtrain, Ytrain)
Ypredict = clf.predict(Xcheck)
print("accuracy: ", metrics.accuracy_score(Ycheck, Ypredict))

import graphviz
dot_data = tree.export_graphviz(clf, out_file = None, feature_names = Dataframe.columns[:-1],
                               class_names = ["Negative", "Positive"])

graph = graphviz.Source(dot_data)
graph
# End of code
```

3. 의사 결정 트리 시각화 이미지



4. 학습에 활용된 데이터 세트 전체

Dataset

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	1	145	233	1	2	150	0	2.3	3	0	fixed	0
67	1	4	160	286	0	2	108	1	1.5	2	3	normal	1
67	1	4	120	229	0	2	129	1	2.6	2	2	reversible	0
37	1	3	130	250	0	0	187	0	3.5	3	0	normal	0
41	0	2	130	204	0	2	172	0	1.4	1	0	normal	0
56	1	2	120	236	0	0	178	0	0.8	1	0	normal	0
62	0	4	140	268	0	2	160	0	3.6	3	2	normal	1
57	0	4	120	354	0	0	163	1	0.6	1	0	normal	0
63	1	4	130	254	0	2	147	0	1.4	2	1	reversible	1
53	1	4	140	203	1	2	155	1	3.1	3	0	reversible	0
57	1	4	140	192	0	0	148	0	0.4	2	0	fixed	0
56	0	2	140	294	0	2	153	0	1.3	2	0	normal	0
56	1	3	130	256	1	2	142	1	0.6	2	1	fixed	1
44	1	2	120	263	0	0	173	0	0	1	0	reversible	0
52	1	3	172	199	1	0	162	0	0.5	1	0	reversible	0
57	1	3	150	168	0	0	174	0	1.6	1	0	normal	0
48	1	2	110	229	0	0	168	0	1	3	0	reversible	0
54	1	4	140	239	0	0	160	0	1.2	1	0	normal	0
48	0	3	130	275	0	0	139	0	0.2	1	0	normal	0
49	1	2	130	266	0	0	171	0	0.6	1	0	normal	0
64	1	1	110	211	0	2	144	1	1.8	2	0	normal	0
58	0	1	150	283	1	2	162	0	1	1	0	normal	0
58	1	2	120	284	0	2	160	0	1.8	2	0	normal	0
58	1	3	132	224	0	2	173	0	3.2	1	2	reversible	1
60	1	4	130	206	0	2	132	1	2.4	2	2	reversible	1
50	0	3	120	219	0	0	158	0	1.6	2	0	normal	0
58	0	3	120	340	0	0	172	0	0	1	0	normal	0
66	0	1	150	226	0	0	114	0	2.6	3	0	normal	0
43	1	4	150	247	0	0	171	0	1.5	1	0	normal	0
40	1	4	110	167	0	2	114	1	2	2	0	reversible	1
69	0	1	140	239	0	0	151	0	1.8	1	2	normal	0
60	1	4	117	230	1	0	160	1	1.4	1	2	reversible	1
64	1	3	140	335	0	0	158	0	0	1	0	normal	0
59	1	4	135	234	0	0	161	0	0.5	2	0	reversible	0

44	1	3	130	233	0	0	179	1	0.4	1	0	normal	0
42	1	4	140	226	0	0	178	0	0	1	0	normal	0
43	1	4	120	177	0	2	120	1	2.5	2	0	reversible	1
57	1	4	150	276	0	2	112	1	0.6	2	1	fixed	0
55	1	4	132	353	0	0	132	1	1.2	2	1	reversible	1
61	1	3	150	243	1	0	137	1	1	2	0	normal	0
65	0	4	150	225	0	2	114	0	1	2	3	reversible	1
65	0	3	155	269	0	0	148	0	0.8	1	0	normal	0
67	1	4	125	254	1	0	163	0	0.2	2	2	reversible	1
62	1	4	120	267	0	0	99	1	1.8	2	2	reversible	0
65	1	4	110	248	0	2	158	0	0.6	1	2	fixed	0
44	1	4	110	197	0	2	177	0	0	1	1	normal	0
65	0	3	160	360	0	2	151	0	0.8	1	0	normal	0
60	1	4	125	258	0	2	141	1	2.8	2	1	reversible	0
51	0	3	140	308	0	2	142	0	1.5	1	1	normal	0
48	1	2	130	245	0	2	180	0	0.2	2	0	normal	0
58	1	4	150	270	0	2	111	1	0.8	1	0	reversible	1
45	1	4	104	208	0	2	148	1	3	2	0	normal	0
53	0	4	130	264	0	2	143	0	0.4	2	0	normal	0
39	1	3	140	321	0	2	182	0	0	1	0	normal	0
68	1	3	180	274	1	2	150	1	1.6	2	0	reversible	1
52	1	2	120	325	0	0	172	0	0.2	1	0	normal	0
44	1	3	140	235	0	2	180	0	0	1	0	normal	0
47	1	3	138	257	0	2	156	0	0	1	0	normal	0
53	0	4	138	234	0	2	160	0	0	1	0	normal	0
51	0	3	130	256	0	2	149	0	0.5	1	0	normal	0
66	1	4	120	302	0	2	151	0	0.4	2	0	normal	0
62	0	4	160	164	0	2	145	0	6.2	3	3	reversible	1
62	1	3	130	231	0	0	146	0	1.8	2	3	reversible	0
44	0	3	108	141	0	0	175	0	0.6	2	0	normal	0
63	0	3	135	252	0	2	172	0	0	1	0	normal	0
52	1	4	128	255	0	0	161	1	0	1	1	reversible	0
62	0	3	130	263	0	0	97	0	1.2	2	1	reversible	1
41	1	2	135	203	0	0	132	0	0	2	0	fixed	0
58	1	3	140	211	1	2	165	0	0	1	0	normal	0
35	0	4	138	183	0	0	182	0	1.4	1	0	normal	0
63	1	4	130	330	1	2	132	1	1.8	1	3	reversible	1
51	1	3	100	222	0	0	143	1	1.2	2	0	normal	0

55	1	4	140	217	0	0	111	1	5.6	3	0	reversible	1
65	1	1	138	282	1	2	174	0	1.4	2	1	normal	0
45	0	2	130	234	0	2	175	0	0.6	2	0	normal	0
56	0	4	200	288	1	2	133	1	4	3	2	reversible	1
54	1	4	110	239	0	0	126	1	2.8	2	1	reversible	1
44	1	2	120	220	0	0	170	0	0	1	0	normal	0
62	0	4	124	209	0	0	163	0	0	1	0	normal	0
54	1	3	120	258	0	2	147	0	0.4	2	0	reversible	0
51	1	3	94	227	0	0	154	1	0	1	1	reversible	0
29	1	2	130	204	0	2	202	0	0	1	0	normal	0
51	1	4	140	261	0	2	186	1	0	1	0	normal	0
43	0	3	122	213	0	0	165	0	0.2	2	0	normal	0
55	0	2	135	250	0	2	161	0	1.4	2	0	normal	0
70	1	4	145	174	0	0	125	1	2.6	3	0	reversible	1
62	1	2	120	281	0	2	103	0	1.4	2	1	reversible	1
35	1	4	120	198	0	0	130	1	1.6	2	0	reversible	0
51	1	3	125	245	1	2	166	0	2.4	2	0	normal	0
59	1	2	140	221	0	0	164	1	0	1	0	normal	0
59	1	1	170	288	0	2	159	0	0.2	2	0	reversible	0
52	1	2	128	205	1	0	184	0	0	1	0	normal	0
64	1	3	125	309	0	0	131	1	1.8	2	0	reversible	0
58	1	3	105	240	0	2	154	1	0.6	2	0	reversible	0
47	1	3	108	243	0	0	152	0	0	1	0	normal	0
57	1	4	165	289	1	2	124	0	1	2	3	reversible	1
41	1	3	112	250	0	0	179	0	0	1	0	normal	0
45	1	2	128	308	0	2	170	0	0	1	0	normal	0
60	0	3	102	318	0	0	160	0	0	1	1	normal	0
52	1	1	152	298	1	0	178	0	1.2	2	0	reversible	0
42	0	4	102	265	0	2	122	0	0.6	2	0	normal	0
67	0	3	115	564	0	2	160	0	1.6	2	0	reversible	0
55	1	4	160	289	0	2	145	1	0.8	2	1	reversible	1
64	1	4	120	246	0	2	96	1	2.2	3	1	normal	1
70	1	4	130	322	0	2	109	0	2.4	2	3	normal	0
51	1	4	140	299	0	0	173	1	1.6	1	0	reversible	0
58	1	4	125	300	0	2	171	0	0	1	2	reversible	0
60	1	4	140	293	0	2	170	0	1.2	2	2	reversible	1
68	1	3	118	277	0	0	151	0	1	1	1	reversible	0
46	1	2	101	197	1	0	156	0	0	1	0	reversible	0

77	1	4	125	304	0	2	162	1	0	1	3	normal	1
54	0	3	110	214	0	0	158	0	1.6	2	0	normal	0
58	0	4	100	248	0	2	122	0	1	2	0	normal	0
48	1	3	124	255	1	0	175	0	0	1	2	normal	0
57	1	4	132	207	0	0	168	1	0	1	0	reversible	0
54	0	2	132	288	1	2	159	1	0	1	1	normal	0
35	1	4	126	282	0	2	156	1	0	1	0	reversible	0
45	0	2	112	160	0	0	138	0	0	2	0	normal	0
70	1	3	160	269	0	0	112	1	2.9	2	1	reversible	1
53	1	4	142	226	0	2	111	1	0	1	0	reversible	0
59	0	4	174	249	0	0	143	1	0	2	0	normal	0
62	0	4	140	394	0	2	157	0	1.2	2	0	normal	0
64	1	4	145	212	0	2	132	0	2	2	2	fixed	1
57	1	4	152	274	0	0	88	1	1.2	2	1	reversible	0
52	1	4	108	233	1	0	147	0	0.1	1	3	reversible	0
56	1	4	132	184	0	2	105	1	2.1	2	1	fixed	0
43	1	3	130	315	0	0	162	0	1.9	1	1	normal	0
53	1	3	130	246	1	2	173	0	0	1	3	normal	0
48	1	4	124	274	0	2	166	0	0.5	2	0	reversible	1
56	0	4	134	409	0	2	150	1	1.9	2	2	reversible	1
42	1	1	148	244	0	2	178	0	0.8	1	2	normal	0
59	1	1	178	270	0	2	145	0	4.2	3	0	reversible	0
60	0	4	158	305	0	2	161	0	0	1	0	normal	0
63	0	2	140	195	0	0	179	0	0	1	2	normal	0
42	1	3	120	240	1	0	194	0	0.8	3	0	reversible	0
66	1	2	160	246	0	0	120	1	0	2	3	fixed	1
54	1	2	192	283	0	2	195	0	0	1	1	reversible	0
69	1	3	140	254	0	2	146	0	2	2	3	reversible	1
50	1	3	129	196	0	0	163	0	0	1	0	normal	0
51	1	4	140	298	0	0	122	1	4.2	2	3	reversible	1
62	0	4	138	294	1	0	106	0	1.9	2	3	normal	1
68	0	3	120	211	0	2	115	0	1.5	2	0	normal	0
67	1	4	100	299	0	2	125	1	0.9	2	2	normal	1
69	1	1	160	234	1	2	131	0	0.1	2	1	normal	0
45	0	4	138	236	0	2	152	1	0.2	2	0	normal	0
50	0	2	120	244	0	0	162	0	1.1	1	0	normal	0
59	1	1	160	273	0	2	125	0	0	1	0	normal	0
50	0	4	110	254	0	2	159	0	0	1	0	normal	0

64	0	4	180	325	0	0	154	1	0	1	0	normal	0
57	1	3	150	126	1	0	173	0	0.2	1	1	reversible	0
64	0	3	140	313	0	0	133	0	0.2	1	0	reversible	0
43	1	4	110	211	0	0	161	0	0	1	0	reversible	0
45	1	4	142	309	0	2	147	1	0	2	3	reversible	1
58	1	4	128	259	0	2	130	1	3	2	2	reversible	1
50	1	4	144	200	0	2	126	1	0.9	2	0	reversible	1
55	1	2	130	262	0	0	155	0	0	1	0	normal	0
62	0	4	150	244	0	0	154	1	1.4	2	0	normal	0
37	0	3	120	215	0	0	170	0	0	1	0	normal	0
38	1	1	120	231	0	0	182	1	3.8	2	0	reversible	1
41	1	3	130	214	0	2	168	0	2	2	0	normal	0
66	0	4	178	228	1	0	165	1	1	2	2	reversible	1
52	1	4	112	230	0	0	160	0	0	1	1	normal	0
56	1	1	120	193	0	2	162	0	1.9	2	0	reversible	0
46	0	2	105	204	0	0	172	0	0	1	0	normal	0
46	0	4	138	243	0	2	152	1	0	2	0	normal	0
64	0	4	130	303	0	0	122	0	2	2	2	normal	0
59	1	4	138	271	0	2	182	0	0	1	0	normal	0
41	0	3	112	268	0	2	172	1	0	1	0	normal	0
54	0	3	108	267	0	2	167	0	0	1	0	normal	0
39	0	3	94	199	0	0	179	0	0	1	0	normal	0
53	1	4	123	282	0	0	95	1	2	2	2	reversible	1
63	0	4	108	269	0	0	169	1	1.8	2	2	normal	0
34	0	2	118	210	0	0	192	0	0.7	1	0	normal	0
47	1	4	112	204	0	0	143	0	0.1	1	0	normal	0
67	0	3	152	277	0	0	172	0	0	1	1	normal	0
54	1	4	110	206	0	2	108	1	0	2	1	normal	1
66	1	4	112	212	0	2	132	1	0.1	1	1	normal	1
52	0	3	136	196	0	2	169	0	0.1	2	0	normal	0
55	0	4	180	327	0	1	117	1	3.4	2	0	normal	1
49	1	3	118	149	0	2	126	0	0.8	1	3	normal	0
74	0	2	120	269	0	2	121	1	0.2	1	1	normal	0
54	0	3	160	201	0	0	163	0	0	1	1	normal	0
54	1	4	122	286	0	2	116	1	3.2	2	2	normal	1
56	1	4	130	283	1	2	103	1	1.6	3	0	reversible	1
46	1	4	120	249	0	2	144	0	0.8	1	0	reversible	0
49	0	2	134	271	0	0	162	0	0	2	0	normal	0

42	1	2	120	295	0	0	162	0	0	1	0	normal	0
41	1	2	110	235	0	0	153	0	0	1	0	normal	0
41	0	2	126	306	0	0	163	0	0	1	0	normal	0
49	0	4	130	269	0	0	163	0	0	1	0	normal	0
61	1	1	134	234	0	0	145	0	2.6	2	2	normal	1
60	0	3	120	178	1	0	96	0	0	1	0	normal	0
67	1	4	120	237	0	0	71	0	1	2	0	normal	1
58	1	4	100	234	0	0	156	0	0.1	1	1	reversible	1
47	1	4	110	275	0	2	118	1	1	2	1	normal	0
52	1	4	125	212	0	0	168	0	1	1	2	reversible	1
62	1	2	128	208	1	2	140	0	0	1	0	normal	0
57	1	4	110	201	0	0	126	1	1.5	2	0	fixed	0
58	1	4	146	218	0	0	105	0	2	2	1	reversible	0
64	1	4	128	263	0	0	105	1	0.2	2	1	reversible	0
51	0	3	120	295	0	2	157	0	0.6	1	0	normal	0
43	1	4	115	303	0	0	181	0	1.2	2	0	normal	0
42	0	3	120	209	0	0	173	0	0	2	0	normal	0
67	0	4	106	223	0	0	142	0	0.3	1	2	normal	0
76	0	3	140	197	0	1	116	0	1.1	2	0	normal	0
70	1	2	156	245	0	2	143	0	0	1	0	normal	0
57	1	2	124	261	0	0	141	0	0.3	1	0	reversible	0
44	0	3	118	242	0	0	149	0	0.3	2	1	normal	0
58	0	2	136	319	1	2	152	0	0	1	2	normal	1
60	0	1	150	240	0	0	171	0	0.9	1	0	normal	0
44	1	3	120	226	0	0	169	0	0	1	0	normal	0
61	1	4	138	166	0	2	125	1	3.6	2	1	normal	1
42	1	4	136	315	0	0	125	1	1.8	2	0	fixed	1
59	1	3	126	218	1	0	134	0	2.2	2	1	fixed	1
40	1	4	152	223	0	0	181	0	0	1	0	reversible	0
42	1	3	130	180	0	0	150	0	0	1	0	normal	0
61	1	4	140	207	0	2	138	1	1.9	1	1	reversible	0
66	1	4	160	228	0	2	138	0	2.3	1	0	fixed	0
46	1	4	140	311	0	0	120	1	1.8	2	2	reversible	1
71	0	4	112	149	0	0	125	0	1.6	2	0	normal	0
59	1	1	134	204	0	0	162	0	0.8	1	2	normal	0
64	1	1	170	227	0	2	155	0	0.6	2	0	reversible	0
66	0	3	146	278	0	2	152	0	0	2	1	normal	0
39	0	3	138	220	0	0	152	0	0	2	0	normal	0

57	1	2	154	232	0	2	164	0	0	1	1	normal	0
58	0	4	130	197	0	0	131	0	0.6	2	0	normal	0
57	1	4	110	335	0	0	143	1	3	2	1	reversible	1
47	1	3	130	253	0	0	179	0	0	1	0	normal	0
55	0	4	128	205	0	1	130	1	2	2	1	reversible	1
35	1	2	122	192	0	0	174	0	0	1	0	normal	0
61	1	4	148	203	0	0	161	0	0	1	1	reversible	1
58	1	4	114	318	0	1	140	0	4.4	3	3	fixed	1
58	0	4	170	225	1	2	146	1	2.8	2	2	fixed	1
56	1	2	130	221	0	2	163	0	0	1	0	reversible	0
56	1	2	120	240	0	0	169	0	0	3	0	normal	0
67	1	3	152	212	0	2	150	0	0.8	2	0	reversible	0
55	0	2	132	342	0	0	166	0	1.2	1	0	normal	0
44	1	4	120	169	0	0	144	1	2.8	3	0	fixed	1
63	1	4	140	187	0	2	144	1	4	1	2	reversible	1
63	0	4	124	197	0	0	136	1	0	2	0	normal	0
41	1	2	120	157	0	0	182	0	0	1	0	normal	0
59	1	4	164	176	1	2	90	0	1	2	2	fixed	1
57	0	4	140	241	0	0	123	1	0.2	2	0	reversible	0
45	1	1	110	264	0	0	132	0	1.2	2	0	reversible	0
68	1	4	144	193	1	0	141	0	3.4	2	2	reversible	1
57	1	4	130	131	0	0	115	1	1.2	2	1	reversible	1
57	0	2	130	236	0	2	174	0	0	2	1	normal	0
59	1	0	164	176	1	0	90	0	1	1	2	1	0
57	0	0	140	241	0	1	123	1	0.2	1	0	normal	0
45	1	3	110	264	0	1	132	0	1.2	1	0	normal	0
68	1	0	144	193	1	1	141	0	3.4	1	2	normal	0
57	1	0	130	131	0	1	115	1	1.2	1	1	normal	0
57	0	1	130	236	0	0	174	0	0	1	1	2	0
40	1	1	140	199	0	0	178	1	1.4	1	0	reversible	0
71	0	2	160	302	0	0	162	0	0.4	1	2	normal	0
59	1	3	150	212	1	0	157	0	1.6	1	0	normal	0
61	0	4	130	330	0	2	169	0	0	1	0	normal	0
58	1	3	112	230	0	2	165	0	2.5	2	1	reversible	1
51	1	3	110	175	0	0	123	0	0.6	1	0	normal	0
50	1	4	150	243	0	2	128	0	2.6	2	0	reversible	1
65	0	3	140	417	1	2	157	0	0.8	1	1	normal	0
53	1	3	130	197	1	2	152	0	1.2	3	0	normal	0

41	0	2	105	198	0	0	168	0	0	1	1	normal	0
65	1	4	120	177	0	0	140	0	0.4	1	0	reversible	0
44	1	4	112	290	0	2	153	0	0	1	1	normal	1
44	1	2	130	219	0	2	188	0	0	1	0	normal	0
60	1	4	130	253	0	0	144	1	1.4	1	1	reversible	0
54	1	4	124	266	0	2	109	1	2.2	2	1	reversible	0
50	1	3	140	233	0	0	163	0	0.6	2	1	reversible	0
41	1	4	110	172	0	2	158	0	0	1	0	reversible	0
54	1	3	125	273	0	2	152	0	0.5	3	1	normal	0
51	1	1	125	213	0	2	125	1	1.4	1	1	normal	0
51	0	4	130	305	0	0	142	1	1.2	2	0	reversible	1
46	0	3	142	177	0	2	160	1	1.4	3	0	normal	0
58	1	4	128	216	0	2	131	1	2.2	2	3	reversible	0
54	0	3	135	304	1	0	170	0	0	1	0	normal	0
54	1	4	120	188	0	0	113	0	1.4	2	1	reversible	1
60	1	4	145	282	0	2	142	1	2.8	2	2	reversible	1
60	1	3	140	185	0	2	155	0	3	2	0	normal	0
54	1	3	150	232	0	2	165	0	1.6	1	0	reversible	0
59	1	4	170	326	0	2	140	1	3.4	3	0	reversible	1
46	1	3	150	231	0	0	147	0	3.6	2	0	normal	0
59	1	4	110	239	0	2	142	1	1.2	2	1	reversible	1
60	0	4	150	258	0	2	157	0	2.6	2	2	reversible	1
52	1	2	134	201	0	0	158	0	0.8	1	1	normal	0
48	1	4	122	222	0	2	186	0	0	1	0	normal	0
45	1	4	115	260	0	2	185	0	0	1	0	normal	0
34	1	1	118	182	0	2	174	0	0	1	0	normal	0
57	0	4	128	303	0	2	159	0	0	1	1	normal	0
71	0	3	110	265	1	2	130	0	0	1	1	normal	0
49	1	3	120	188	0	0	139	0	2	2	3	reversible	1
54	1	2	108	309	0	0	156	0	0	1	0	reversible	0
59	1	4	140	177	0	0	162	1	0	1	1	reversible	1
57	1	3	128	229	0	2	150	0	0.4	2	1	reversible	0
61	1	4	120	260	0	0	140	1	3.6	2	1	reversible	1
39	1	4	118	219	0	0	140	0	1.2	2	0	reversible	1
61	0	4	145	307	0	2	146	1	1	2	0	reversible	0
56	1	4	125	249	1	2	144	1	1.2	2	1	normal	0
52	1	1	118	186	0	2	190	0	0	2	0	fixed	0
43	0	4	132	341	1	2	136	1	3	2	0	reversible	1

65	1	4	135	254	0	2	127	0	2.8	2	1	reversible	1
48	1	4	130	256	1	2	150	1	0	1	2	reversible	1
63	0	4	150	407	0	2	154	0	4	2	3	reversible	1