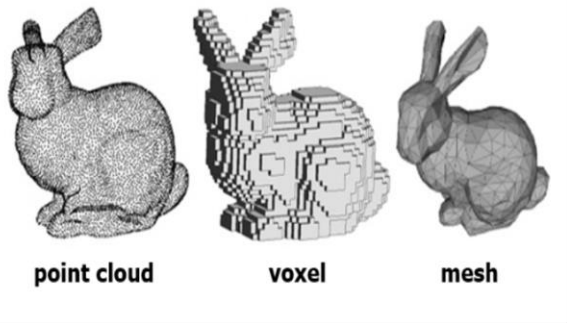
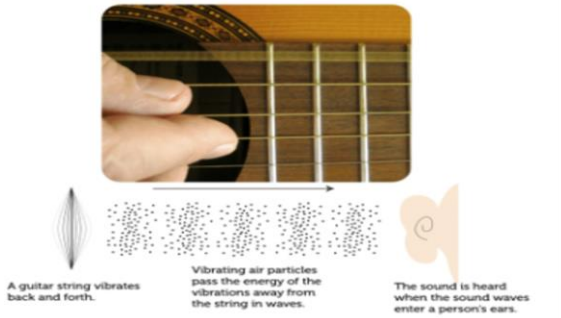
[기계학습] 텀프로젝트 안내

텀프로젝트 개요



3D 데이터

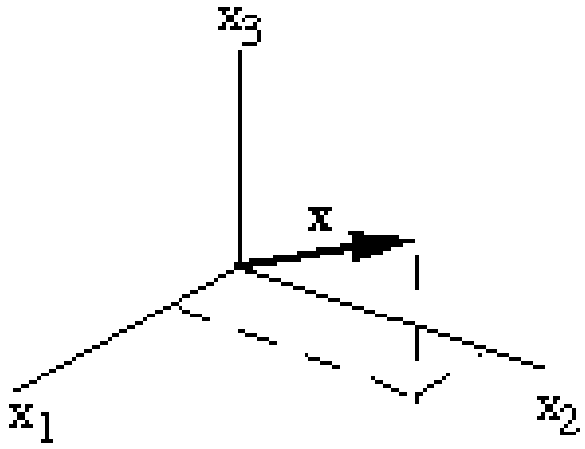
2D 데이터



음성 데이터

텍스트 데이터

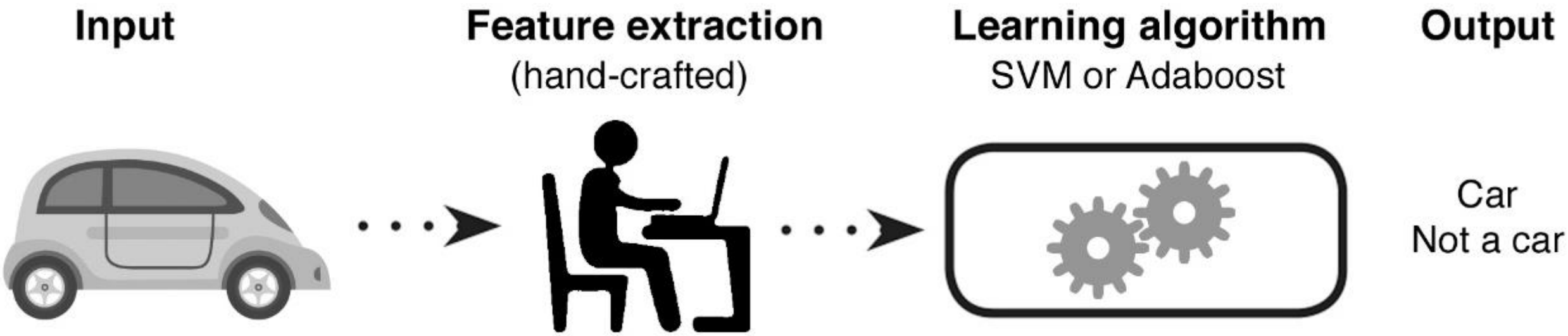
$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_d \end{bmatrix}$$



특징 벡터
(feature vector)

복잡한 비정형 데이터들에 대해 기계학습 방법론을 사용하기 위해서
어떻게 특징 벡터(feature vector)로 변화할 수 있을까?

텀프로젝트 개요



다양한 도메인의 데이터의 특징을 효과적으로 추출할 수 있는
전통적인 Hand-Crafted Feature Extraction 방법을 배울 수 있다.

텀프로젝트 개요

```python
# sklearn의 LogisticRegression 활용하여 회귀 계수 구해보기
from sklearn.linear_model import LogisticRegression

logistic_reg = LogisticRegression()
logistic_reg.fit(X, y)
y_predict = lin_reg.predict(X_new)
```

```
class LogisticRegression(LinearClassifierMixin, SparseCoefMixin, BaseEstimator):
    """
    Logistic Regression (aka logit, MaxEnt) classifier.

    In the multiclass case, the training algorithm uses the one-vs-rest (OvR)
    scheme if the 'multi_class' option is set to 'ovr', and uses the
    cross-entropy loss if the 'multi_class' option is set to 'multinomial'.
    (Currently the 'multinomial' option is supported only by the 'lbfgs',
    'sag', 'saga' and 'newton-cg' solvers.)

    This class implements regularized logistic regression using the
    'liblinear' library, 'newton-cg', 'sag', 'saga' and 'lbfgs' solvers. **Note
    that regularization is applied by default**. It can handle both dense
    and sparse input. Use C-ordered arrays or CSR matrices containing 64-bit
    floats for optimal performance; any other input format will be converted
    (and copied).

    The 'newton-cg', 'sag', and 'lbfgs' solvers support only L2 regularization
    with primal formulation, or no regularization. The 'liblinear' solver
```

SciKit-Learn만을 단순히 활용하는 것이 아니라 직접 알고리즘을 구현함으로써
기계학습 방법론의 이해도를 더욱 높일 수 있다.

텀프로젝트 안내

프로젝트 문제

Feature Engineering

- [1D 텍스트 데이터] 한국어 영화 리뷰 분류하기
- [2D 영상 데이터] 2D 이미지 데이터를 활용한 이미지 분류
- [3D 영상 데이터] 3D 데이터를 활용한 물체 분류
- [1D 음성 데이터] 음악 장르 분류

Algorithm Implementation

- [선형모델] 경사하강법 구현
- [앙상블] 의사결정나무 구현

Feature Engineering

[음성/텍스트]

- 한국어 영화 리뷰 분류하기
- 음악 장르 분류하기

[2D 이미지/3D 데이터]

- 2D 이미지 데이터 활용한 이미지 분류
- 3D 데이터를 활용한 물체 분류

[음성/텍스트]중 에서 하나 선택
[2D 이미지/3D 데이터]중 에서 하나 선택

총 두 문제를 풀어야 합니다.

텀프로젝트 안내

프로젝트 문제

Feature Engineering

- [1D 텍스트 데이터] 한국어 영화 리뷰 분류하기
- [2D 영상 데이터] 2D 이미지 데이터를 활용한 이미지 분류
- [3D 영상 데이터] 3D 데이터를 활용한 물체 분류
- [1D 음성 데이터] 음악 장르 분류

Algorithm Implementation

- [선형모델] 경사하강법 구현
- [앙상블] 의사결정나무 구현

Algorithm Implementation

[선형모델]

- 경사하강법 구현

[앙상블]

- 의사결정나무 구현

<u>총 두 문제를 풀어야 합니다.</u>

텀프로젝트 진행방식 – 코드

```
def Construct_Subtree(node, max_depth):
    if (node.depth == max_depth or len(node.y) == 1): # node의 깊이가 max_depth에 도달했거나 리프 노드일 때
        ####### Empty Module.4 #######
        node.predictor =                    # node 내부에 있는 y값들의 평균을 활용하여 예측 수행
        ##############################
    else:
        j, xi = CalculateOptimalSplit(node)
        node.j = j
        node.xi = xi
        Xt, yt, Xf, yf = DataSplit(node.X, node.y, j, xi)

        if (len(yt)>0):
            ####### Empty Module.5 #######
            node.left =                             # TNode를 활용하여 새로운 왼쪽 자식 노드 구축
            Construct_Subtree(,)                    # Construct_Subtree를 활용하여 왼쪽 자식 노드에 대한 Subtree 구축
            ##############################
        if (len(yf)>0):
            ####### Empty Module.6 #######
            node.right =                            # TNode를 활용하여 새로운 오른쪽 자식 노드 구축
            Construct_Subtree(,)                    # Construct_Subtree를 활용하여 오른쪽 자식 노드에 대한 Subtree 구축
            ##############################

    return node
```

각 문제마다 Empty Module을 안내사항에 따라 채우면 됩니다.

텀프로젝트 진행방식 – 해설 영상

- **Feature Engineering** 문제 해결 과정을 담은 해설 영상을 녹화해서 제출하시면 됩니다.
  - 발표 PPT는 자유 양식이며, 비디오 영상 녹화본 길이의 제한은 없습니다.
  - 해설 영상 예시 : 링크

텀프로젝트 진행방식 – 보고서

보고서

1. 사이킷런에 구현되어 있는 Linear Regression은 이번 텀프로젝트에서 구현한 경사 하강법(Gradient Descent) 기반의 방식과 어떤 차이가 있는지 자유롭게 서술하세요. [1점]
    - 회귀 계수를 추정하는 과정에서의 다른 부분을 위주로 서술하시면 됩니다.
2. 사이킷런에 구현되어 있는 Logistic Regression은 이번 텀프로젝트에서 구현한 경사 하강법(Gradient Descent) 기반의 방식과 어떤 차이가 있는지 자유롭게 서술하세요. [1점]
    - 회귀 계수를 추정하는 과정에서의 다른 부분을 위주로 서술하시면 됩니다.

보고서

1. 사이킷런에 구현되어 있는 DecisionTreeRegressor은 이번 텀프로젝트에서 직접 구현한 `1. 결정 트리(Decision Tree)` 방식과 어떤 차이가 있는지 자유롭게 서술하세요. [1점]
    - 파라미터 관점에서 기능이 다른 부분을 위주로 서술하시면 됩니다.
2. 사이킷런에 구현되어 있는 RandomForestRegressor은 이번 텀프로젝트에서 구현한 `2. 배깅(Bagging)` 방식과 어떤 차이가 있는지 자유롭게 서술하세요. [1점]
    - 의사 결정 나무를 구성하는 과정에서 다른 부분을 위주로 서술하시면 됩니다.

텀프로젝트 마감

프로젝트 마감

- 리더보드 마감: 6월 4일 (일) 오후 11시 59분
- 자료 마감: 6월 5일 (월) 오후 11시 59분
  - Feature Engineering
    - 문제 마다 발표자료 및 발표 영상
    - 캐글 노트북 공유
  - Algorithm Implementation
    - 문제마다 ipynb 파일 제출 (주석 상세하게 적어야 함)
    - 문제마다 제공된 보고서 제출

텀프로젝트 배점 방식 (중요)

배점 방식 (총 20점 만점)

- Feature Engineering
  - 문제마다 5점 (총 10점)
- Algorithm Implementation
  - 문제마다 5점 (총 10점)

추가 배점 방식

- **Feature Engineering** 문제의 경우 **캐글 리더보드 상위 랭커 4인에게 추가 점수 1점**을 부여합니다.
  - 텀프로젝트 20점을 온전하게 받은 상황에서 추가 점수가 발생하는 경우, 추가 점수는 **중간고사/기말고사 점수를 보완할 수 있는 전체 점수**로 부여됩니다.
- 추가 점수를 위해 Feature Engineering 문제를 더 푸는 것도 허용합니다.
  - 만약, 4개의 문제에서 모두 상위 랭커 4인 안에 들었다면 추가 점수 4점 부여
- **추가 점수를 받는 문제에 대해서는 모두 빠짐없이 발표 자료와 발표 영상을 보내야 추가 점수를 부여합니다.**
- 공동 순위자가 4인 이상일 경우 추가 점수 부여하지 않습니다.

텀프로젝트 제출 양식

Feature Engineering 문제 노트북 공유 : [2023-ML][TP][P1]학번_이름 (조교에게 캐글 노트북 공유)

Algorithm Implementation 문제 ipynb 파일 제출 : [2023-ML][TP][P1]학번_이름 (이메일 제출)

발표 자료 제출(PPT) : [2023-ML][발표자료][P1]학번_이름 (이메일 제출)

해설 영상 제출 (MP4): [2023-ML][해설영상][P1]학번_이름 (이메일 제출)

조교들 캐글 노트북 : Lim Guen Taek, sukzoon1234, sjkimhyunwoo, childult
과제 제출 메일 : admin@rcv.sejong.ac.kr