인공지능개론

- 교수명 : 김종유
  - email : jy.kim@yonsei.ac.kr
- 학정번호 : GEV6101-01
- 강의실 : 공D603

25.03.04. (화)

소개

- MS 근무 경력 - Computer Vision 등
- MMAI Lab 활동 (비디오 도메인 쪽)

References

- AI for Beginners (Microsoft github 자료)

수업 정보

- 숙제&프로젝트 10%
- 중간, 기말 각각 40%
- 출결 10%

MS에서 했던 일들

- Visual Perception
- 3D Face Model & Reconstruction

- xbox avatar (3D face modeling 등)

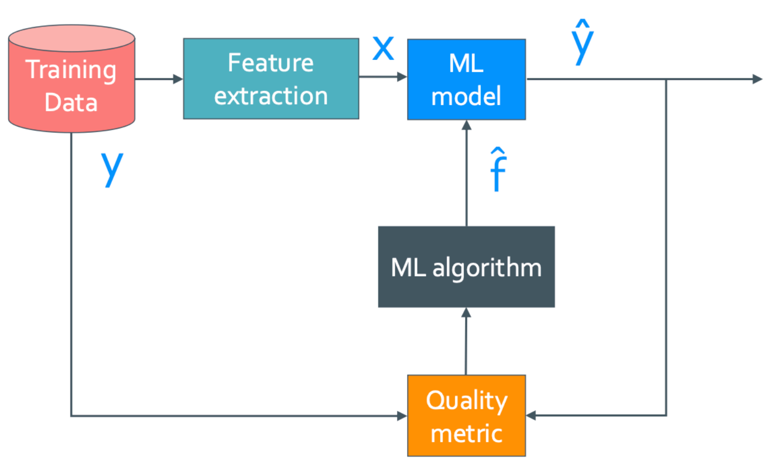
25.03.11. (화)

교수님이 직접 출석을 부름

주제 : Regression

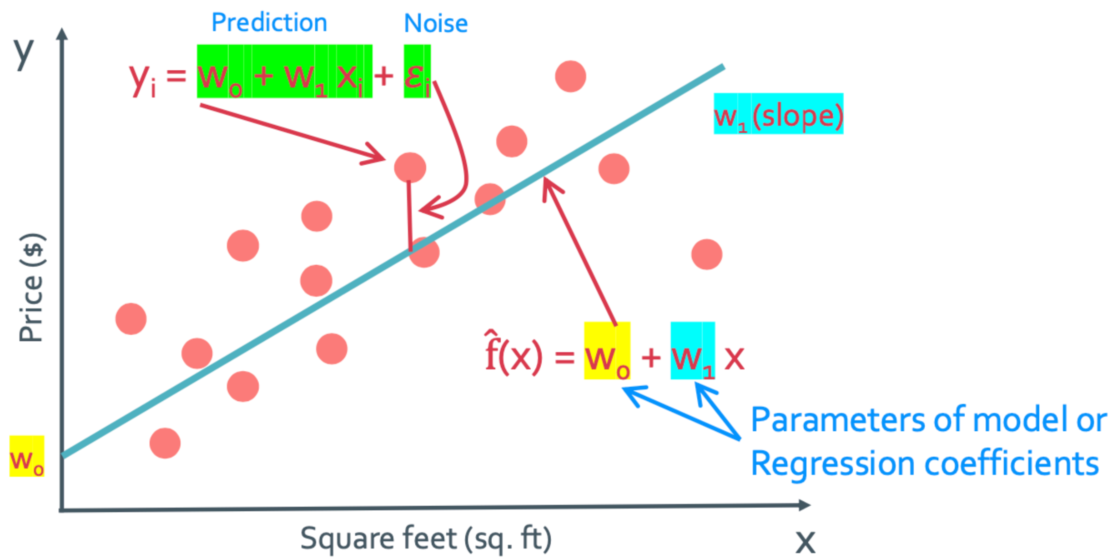
- Regression :
    - A set of statistical processes for estimating the relationships between a dependent variable
    - 변수들을 이용하여 아웃풋의 관계를 모델링 ; Linear Regression이 가장 단순하고 쉬운 방법
- **Linear Regression**
    - Linear regression is the most widely used of all statistical techniques: it is the study
of linear (i.e., straight-line) relationships between variables, usually under an assumption
of normally distributed errors.
    - 가장 간단한 예 : $y = ax + b$
- **Francis Ganlton**
    - 아이들의 키가 부모의 키에 어떤 연관을 가지는지 통계학적 연구
- Linear Regression 이 푸는 문제
    - **House Price vs. Area**

- x = feature (집 넓이) , y = observation (가격 등)
- Model – How we assume the world works
    - Regression model: yi = f(xi) + εi
- 레그리션 모델이 모든 상황에 완벽하게 fit 할 수는 없다. 하지만 best fit을 찾는 것이 머신러닝의 목적
- Linear Regression 의 파이프라인 : Training Data → Feature extraction → ML model ← ML algorithm ← Quality metric



- Linear Regression

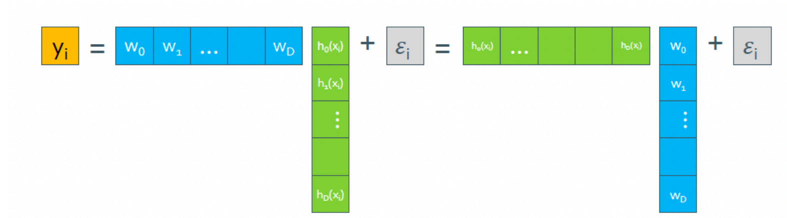
- Use a simple linear regression model
- Fit a line through the data



- "Cost" of using a given line
  - Residual sum of squares (RSS) → 코스트
    - RSS(w0, w1) is a function of 2 variables
- Find "best" line
- Minimizing the cost
- Approach 1: Set gradient = 0
  - **Analytic Solution**
    - For linear regression problem, In particular, we can find the optimal parameters (as assessed on the training data) analytically
    - 이터레이션 없이 수학적으로 풀어낼 수 있는 솔루션
- Approach 2: Follow the slope
- **Approach 2: Gradient descent**
  - Gradient descent relies on choosing step size (learning rate) and convergence criteria
  - **w(t+1) ← w(t) -** η **∇RSS(w(t))**

- Linear Model
  - 파라미터는 2개 뿐, w0, w1
  - *Regression model:*
    - *yi = w0 + w1 xi + εi*
    - *Estimated parameters:*
      ŵ0, ŵ1
- **What about a quadratic function?**
  - Linear 보다는 커브를 줘서 더 잘 예측
- **Even higher order polynomial**
- **Polynomial regression**
  - Model : **yi =w0 + w1 xi + w2 xi2 + …** + wp xip + εi
  - 이 경우 Linear Regression 이냐?
    - x에 대해서는 아니지만, 모델 파라미터 w에 대해서 linear 이냐를 따져야 함. 그래서 이 경우 Linear가 맞음
- Generic basis expansion
  - yi = w0h0(xi) + w1h1(xi) + … + wDhD(xi) + εi
  - = σj=0
    D wjhj(xi) + εi , where h is a feature extractor
  - jth regression coefficient or weight , jth feature
- **Generic linear regression model**
  - 입력이 늘어날 수 있음, featuer도 차원이 늘어날 수 있음
  - **yi = w0h0(xi) + w1h1(xi) + … + wDhD(xi) + εi =** σj=0 D **wjhj(xi) + εi**
  - # observations (**x**i, yi): N
    # inputs
    x[j]: d
    # features hj(
    x): D+1
- **Rewrite in matrix notation**

- Analytics Solution을 구하기
- 선형대수 기본 개념 + 미분에 대한 지식 필요
- **For observation i**
  yi =
  σj=0 D **wjhj(xi) +** ε**i**



- 두 벡터의 내적

- **RSS for multiple regression**
  - RSS(**w**)=σi=1 N (yi-hT(**xi**)**w**)2
    = (
    y-Hw)T(**y-Hw**)
- **Closed-form solution**
  - **∇RSS(w) = -2HT(y-Hw) = 0**
  - **Solve for w:**
    - -2HT(y-Hw) =0
      -2HTy = 2HTHw —> A^-1 * A 로 하면 w가 구해짐
- **Gradient descent**
  - **while not converged**

    w(t+1) ← w(t) - η **∇RSS(w(t))**
    -2HT(y-Hw)
- **Use calculus to compute an analytic gradient!!**

—> 노트 보정

회귀 분석(Regression)의 이해

1. 인공지능과 머신러닝의 관계

인공지능(AI)은 인간의 지능을 모방하는 기계의 능력을 의미하며, 그 중 머신러닝(Machine Learning)은 시스템이 경험을 통해 자동으로 학습하고 개선할 수 있게 하는 AI의 응용 분야입니다. 딥러닝(Deep Learning)은 복잡한 알고리즘과 심층 신경망을 사용하여 모델을 학습시키는 머신러닝의 한 형태입니다.

2. 회귀 분석(Regression)

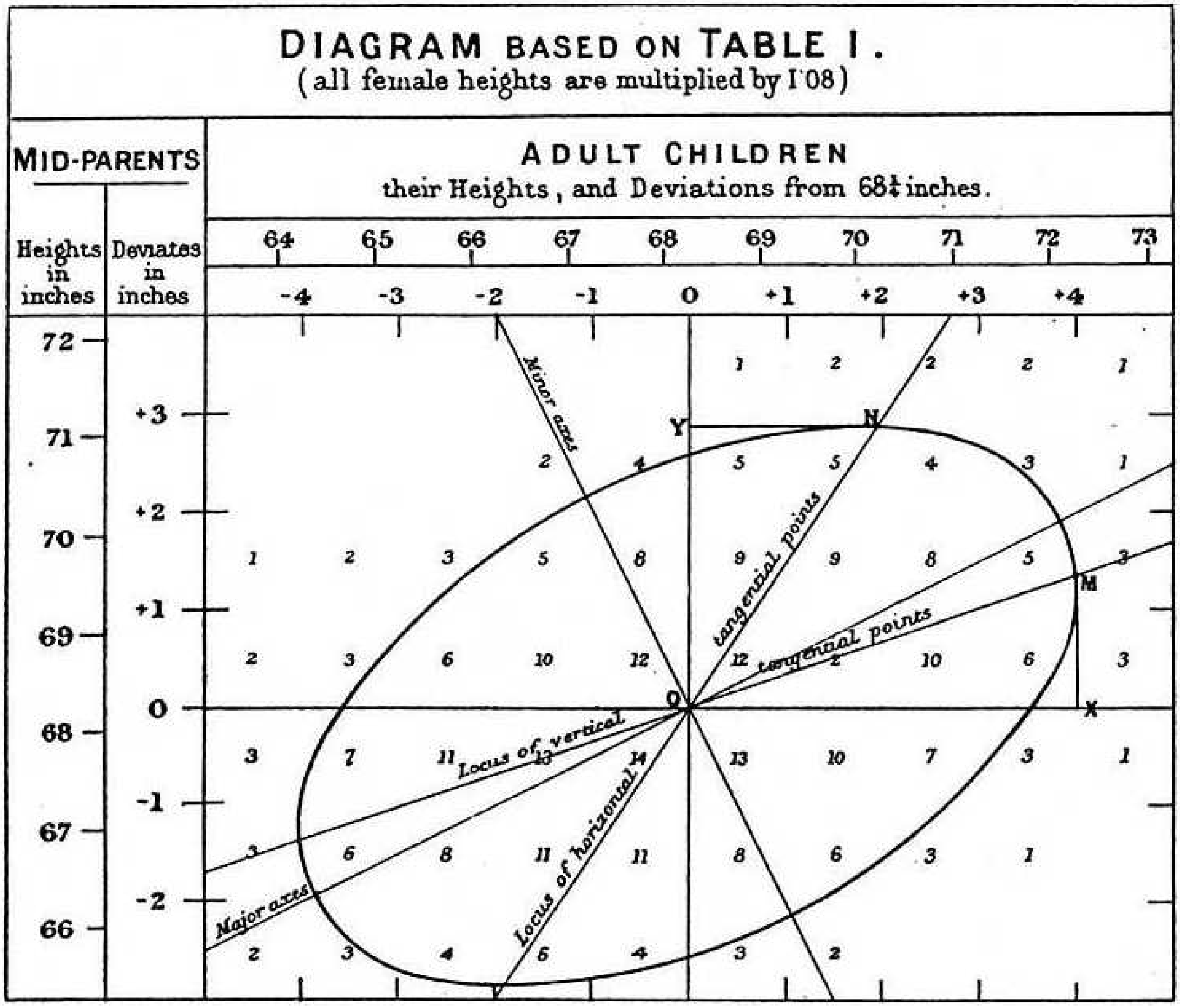
회귀 분석은 종속 변수와 하나 이상의 독립 변수 간의 관계를 추정하는 통계적 프로세스입니다. 그 중에서도 선형 회귀(Linear Regression)는 변수 간의 선형(직선) 관계를 연구하는 가장 널리 사용되는 통계 기법입니다.

선형성(Linearity)의 특성

- **가산성(Additivity)**: f(x + y) = f(x) + f(y)
- **동차성(Homogeneity)**: f(ax) = af(x)
- **예시**: y = ax + b (단순 선형 회귀 방정식)

3. 선형 회귀의 역사적 배경

Francis Galton은 부모와 자녀의 키 사이의 관계를 연구했으며, 이를 통해 "평균으로의 회귀(regression toward the mean)" 현상을 발견했습니다. 이는 극단적인 특성을 가진 부모의 자녀들이 평균에 더 가까운 특성을 보이는 경향을 의미합니다.



4. 선형 회귀 모델 구축

문제 설정: 주택 가격 vs. 면적

주택 가격(y)을 면적(x)을 기반으로 예측하는 모델을 구축해 봅시다.

데이터와 용어

- **x**: 특성(feature) - 예: 주택 면적(sq. ft.)
- **y**: 관측값(observation) - 예: 주택 가격($)

회귀 모델의 기본 형태

$$y_i = f(x_i) + \varepsilon_i$$

여기서:

- y_i는 실제 관측값
- $f(x_i)$는 예측 함수
- ε_i는 오차(noise)

5. 단순 선형 회귀 모델

단순 선형 회귀 모델은 다음과 같이 표현됩니다:

$$\hat{f}(x) = w_0 + w_1 x$$

여기서:

- w_0는 y절편(intercept)
- w_1는 기울기(slope)

6. 비용 함수(Cost Function)

모델의 품질을 평가하기 위해 잔차 제곱합(Residual Sum of Squares, RSS)을 사용합니다:

$$RSS(w_0, w_1) = \Sigma(y_i - [w_0 + w_1 x_i])^2$$

이 함수는 w_0와 w_1의 함수이며, 최적의 파라미터를 찾기 위해 이 비용 함수를 최소화해야 합니다.

7. 최적의 파라미터 찾기

방법 1: 그래디언트를 0으로 설정 (해석적 해법)

선형 회귀 문제에서는 비용 함수를 미분하여 0이 되는 지점을 찾아 최적의 파라미터를 구할 수 있습니다.

방법 2: 경사 하강법(Gradient Descent)

반복적으로 파라미터를 업데이트하여 비용 함수의 최소값을 찾는 방법입니다:

$$w(t+1) \leftarrow w(t) - \eta \nabla RSS(w(t))$$

여기서:

- η는 학습률(learning rate)
- $\nabla RSS(w(t))$는 현재 파라미터에서의 비용 함수의 그래디언트

학습률(η) 선택

- **고정 학습률**: 모든 반복에서 동일한 학습률 사용
- **감소 학습률**: 반복이 진행됨에 따라 학습률 감소 (예: $\eta_t = \alpha/t$ 또는 $\eta_t = \alpha/\sqrt{t}$)

수렴 기준

일반적으로 그래디언트의 크기가 특정 임계값(ε) 미만일 때 수렴했다고 판단합니다:

$$||\nabla RSS(w)|| < \varepsilon$$

8. 다항 회귀(Polynomial Regression)

데이터가 비선형 관계를 보일 때, 다항식을 사용하여 모델링할 수 있습니다:

$$y_i = w_0 + w_1 x_i + w_2 x_i^2 + \ldots + w_p x_i^p + \varepsilon_i$$

이 모델은 x에 대해 비선형이지만, 파라미터 w에 대해서는 선형입니다.

9. 일반적인 기저 확장(Basis Expansion)

더 일반적인 형태로, 다양한 특성 추출기(feature extractor)를 사용할 수 있습니다:

```
yi = w₀h₀(xi) + w₁h₁(xi) + ... + wDhD(xi) + εi = Σⱼ₌₀ᴰ wⱼhⱼ(xi) + εi
```

여기서 h는 특성 추출기 함수입니다.

10. 다중 회귀(Multiple Regression)

여러 입력 변수를 사용하는 회귀 모델:

```
f(x) = w₀ + w₁x[1] + w₂x[2] + ... + wdx[d]
```

여기서:

- x는 d차원 벡터
- x[j]는 j번째 입력 변수

11. 행렬 표기법을 사용한 회귀 모델

다중 회귀 모델은 행렬 표기법으로 간결하게 표현할 수 있습니다:

```
y = Hw + ε
```

여기서:

- y는 관측값 벡터
- H는 디자인 행렬(design matrix)
- w는 파라미터 벡터
- ε는 오차 벡터

닫힌 형태의 해(Closed-form Solution)

```
ŵ = (H^T H)^(-1) H^T y
```

이 해는 H^T H가 가역(invertible)일 때 존재합니다.

12. 수치적 그래디언트 vs. 해석적 그래디언트

- **수치적 그래디언트**: 근사적, 느림, 구현 쉬움
- **해석적 그래디언트**: 정확, 빠름, 오류 발생 가능성 있음

실제로는 해석적 그래디언트를 사용하되, 수치적 그래디언트로 구현을 검증하는 것이 좋습니다 (그래디언트 체크).

요약

이제 여러분은 다음을 할 수 있습니다:

1. 다항 회귀 설명
2. 다중 입력 또는 특성을 사용한 회귀 모델 작성
3. 다항 회귀와 다중 입력 회귀를 다중 특성 회귀로 표현
4. RSS와 같은 적합도 메트릭 계산
5. 일반적인 다중 회귀 모델의 파라미터 추정:
    - 닫힌 형태의 해 사용
    - 반복적 경사 하강법 알고리즘 사용
6. 적합된 함수 해석
7. 예측 형성을 위한 추정 모델 활용