UCLA

CS145 Discussion

Week 2

Junheng, Shengming, Yunsheng

10/12/2018

UCLA

Roadmap

- Reminders
- Review: Linear Regression
  - Examples question
  - Regularization
  - Z-score Normalization
  - Newton's Method
- Review: Logistic Regression
  - Example question
  - Log-likelihood
- Q&A: Course Projects
  - Dataset
  - Some ideas
- Programming Tips

UCLA

Reminders

- Homework 1 released on CCLE, due on 23:59 pm, Oct 19 (next Friday)
  - Programming questions about linear/logistic regression
  - Submit code and report together!
- Course project released on Kaggle
  - Website: https://www.kaggle.com/c/yelpratingprediction
  - Team Sign-up: https://docs.google.com/spreadsheets/d/1psDqf4yTienkOrcI6vJd0BigHRlyCBlRjk6Uf7ki2-M/edit?usp=sharing
  - Important date: Nov.5 - Midterm report, Dec. 9 - Final report
  - Anyone still do not have a team?
- Join Piazza!
  - https://piazza.com/ucla/fall2018/cs145/home

UCLA

Linear Regression: Model

- Linear model to predict value of a variable *y* using features ***x***

$$y = \boldsymbol{x}^T \boldsymbol{\beta} = x_1\beta_1 + x_2\beta_2 + \cdots + x_p\beta_p$$

- Least Square Estimation

$$J(\boldsymbol{\beta}) = \frac{1}{2n}(X\boldsymbol{\beta} - y)^T(X\boldsymbol{\beta} - y)$$

- Closed form solution

$$\hat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T y$$

UCLA

Linear Regression: Example

- A ball is rolled down a hallway and its position is recorded at five different times. Use the table shown below to calculate
  - Weights
  - Predicted position at each given time and at time 12 seconds

Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26

UCLA

Linear Regression: Example

Step 1: Question

- What are X and Y variables?
- What are the parameters for our problem?
- Calculating parameters

Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26

UCLA

Linear Regression: Example

Step 1: Calculate Weights

- What are X and Y variables?
  - Time (X) and Position(Y)
- What are the parameters for our problem?
  - $\hat{\beta}_1$:Time $\hat{\beta}_0$:Intercept
- Calculating parameters
  - $\hat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T y$

Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26

UCLA

Linear Regression: Example

Let's calculate on BOARD!

$$X = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 4 \\ 1 & 6 \\ 1 & 8 \end{bmatrix} \qquad y = \begin{bmatrix} 9 \\ 12 \\ 17 \\ 21 \\ 26 \end{bmatrix}$$

$$X^T X = ? \qquad (X^T X)^{-1} = ?$$

$$X^T y = ? \qquad \hat{\beta} = (X^T X)^{-1} X^T y = ?$$

Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26

Linear Regression: Example

Step 2: Apply your model and predict

- Plug time values into linear regression equation

$$\hat{y} = 2.378x + 7.012$$

- Predicted value at time = 12 secs

$$\hat{y}(x = 12) = 2.378 \times 12 + 7.012 = 35.548$$

- Matrix form to predict all other positions

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\beta}$$

Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26
12	35.55

UCLA

Linear Regression: Example

Plot: Check your model

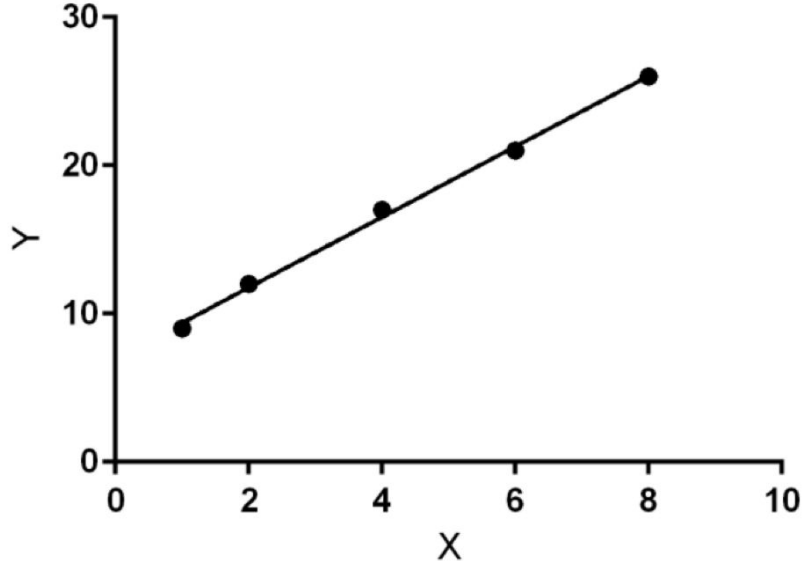
$$\hat{y} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 4 \\ 1 & 6 \\ 1 & 8 \end{bmatrix} \begin{bmatrix} 7.012 \\ 2.378 \end{bmatrix} = \begin{bmatrix} 9.390 \\ 11.768 \\ 16.524 \\ 21.280 \\ 26.036 \end{bmatrix}$$

Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26

UCLA

Linear Regression: Example

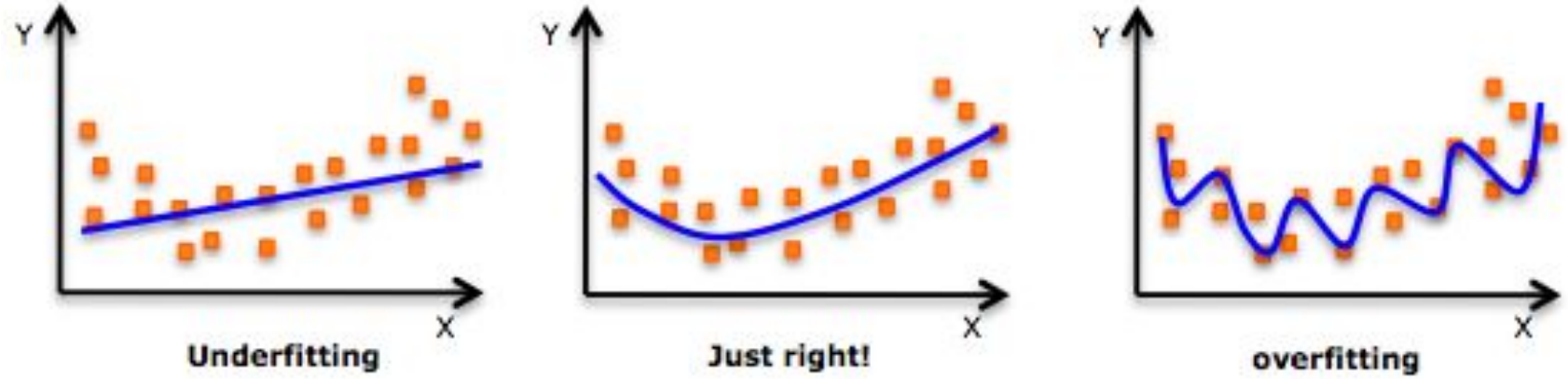
Plot: Check your model



Time (s)	Position (m)
1	9
2	12
4	17
6	21
8	26

Linear Regression + Regularization

- Why do we need regularization?
  - Constraint the parameter values
  - Avoid overfitting phenomenon



Closed form: LR + Regularization

- Prediction

$$\hat{y} = \boldsymbol{x}^T \boldsymbol{\beta} = x_1\beta_1 + x_2\beta_2 + \cdots + x_p\beta_p$$

- Original Objective

$$\min_{\boldsymbol{\beta}} \; J(\boldsymbol{\beta}) = \frac{1}{2}\sum_{i=1}^{N}(\boldsymbol{x}^T\boldsymbol{\beta} - y)^2$$

- L2-Regularized Objective

$$\min_{\boldsymbol{\beta}} \; J(\boldsymbol{\beta}) = \frac{1}{2}\sum_{i=1}^{N}(\boldsymbol{x}^T\boldsymbol{\beta} - y)^2 + \frac{\lambda}{2}||\boldsymbol{\beta}||^2$$

UCLA

Closed form: LR + Regularization

$$\min_{\boldsymbol{\beta}} \; J(\boldsymbol{\beta}) = \frac{1}{2}\sum_{i=1}^{N}(\boldsymbol{x}^T\boldsymbol{\beta} - y)^2 + \frac{\lambda}{2}||\boldsymbol{\beta}||^2$$

$$\frac{\partial J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{N} \boldsymbol{x}(\boldsymbol{x}^T\boldsymbol{\beta} - y) + \frac{\partial \lambda \; ||\boldsymbol{\beta}||^2}{\partial \boldsymbol{\beta}}$$

$$\frac{\partial J(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{N} \boldsymbol{x}(\boldsymbol{x}^T\boldsymbol{\beta} - y) + \lambda\boldsymbol{\beta}$$

$$\nabla_{\boldsymbol{\beta}} J(\boldsymbol{\beta}) = 0$$

$$\nabla_{\boldsymbol{\beta}} \left(\frac{1}{2}\|\boldsymbol{X\beta} - \boldsymbol{y}\|^2 + \frac{\lambda}{2}\|\boldsymbol{\beta}\|^2\right) = 0$$

$$\nabla_{\boldsymbol{\beta}} \left(\frac{1}{2}(\boldsymbol{X\beta} - \boldsymbol{y})^T(\boldsymbol{X\beta} - \boldsymbol{y}) + \frac{\lambda}{2}\boldsymbol{\beta}^T\boldsymbol{\beta}\right) = 0$$

$$\boldsymbol{X}^T\boldsymbol{X\beta} - \boldsymbol{X}^T\boldsymbol{y} + \lambda\boldsymbol{I\beta} = 0$$

$$\boldsymbol{\beta} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

Regularization

Regularization methods

- L2 normalization (Ridge)

$$\min_{\beta} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

- L1 normalization (Lasso)

$$\min_{\beta} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- Dropout (in neural networks)
- Early Stopping

UCLA

Z-Score Normalization

- Why normalize features？
  - Different feature ranges such as [-1, 1] and [-100, 100] may negatively affect algorithm performance.
  - Thus, small change in bigger range can affect more than huge change in smaller range.
- **Calculate Z-Score (Standard Score)**

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

➢ $z_{ij} \rightarrow$ the standard score for feature *j* of data point *i*

➢ $x_{ij} \rightarrow$ the value of feature *j* of data point *i*

➢ $\mu_j, \sigma_j \rightarrow$ the mean and standard deviation of feature *j*

UCLA

Z-Score Normalization: Example

Galaxy	Distance (Mpc)	Velocity (km/sec)
Virgo	15	1,600
Ursa Minor	200	15,000
Corona Borealis	290	24,000
Bootes	520	40,000

- How to Normalize feature “Distance”?

UCLA

Z-Score Normalization: Example

Galaxy	Distance (Mpc)	Velocity (km/sec)
Virgo	15	1,600
Ursa Minor	200	15,000
Corona Borealis	290	24,000
Bootes	520	40,000

- How to Normalize feature “Distance”?
  - Computer mean and deviation first

$$\mu_{\text{dist}} =? \quad \sigma_{\text{dist}} =?$$

UCLA

Z-Score Normalization: Example

Galaxy	Distance (Mpc)	Velocity (km/sec)
Virgo	15	1,600
Ursa Minor	200	15,000
Corona Borealis	290	24,000
Bootes	520	40,000

- How to Normalize feature “Distance”?
  - Computer mean and deviation $\mu_{\text{dist}} = 256.25, \sigma_{\text{dist}} = 181.71$
  - Computer Z-Score for each sample

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

Z-Score Normalization: Example

Galaxy	Distance (Mpc)	Velocity (km/sec)
Virgo	15	1,600
Ursa Minor	200	15,000
Corona Borealis	290	24,000
Bootes	520	40,000

- How to Normalize feature “Distance”?
  - Computer mean and deviation $\mu_{\text{dist}} = 256.25, \sigma_{\text{dist}} = 181.71$
  - Computer Z-Score for each sample (for example “Virgo”)

$$z_{\text{virgo,dist}} = \frac{x_{\text{virgo,dist}} - \mu_{\text{dist}}}{\sigma_{\text{dist}}} = \frac{15 - 256.25}{181.71} = -1.328$$

UCLA

Newton's Method

From lectures:

$$\beta^{\text{new}} = \beta^{\text{old}} - \left(\frac{\partial^2 L(\beta)}{\partial\beta\partial\beta^T}\right)^{-1} \frac{\partial L(\beta)}{\partial\beta}$$

Apply Newton's methods on single variable to find minima:

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}$$

From single variable to Multivariate Newton-Raphson Method

UCLA

Newton's Method: Steps

1. Initialize $x^{(0)}$
2. Calculate $\nabla f(x)$
3. Calculate $F(x)$
4. Initialize step $n = 0$ and start loops
   a. Calculate $\nabla f(x^{(n)})$
   b. Calculate $F(x^{(n)})$
   c. Calculate $[F(x^{(n)})]^{-1}$
   d. Update: $x^{(n+1)} = x^{(n)} - [F(x^{(n)})]^{-1} \cdot \nabla f(x^{(n)})$
   e. Update: $n = n + 1$
5. Exit Loop

UCLA

Newton's Method: Example

$$x^{(0)} = [3, -1, 0]$$

$$f(x_1, x_2, x_3) = (x_1 + 10x_2)^2 + 5(x_1 - x_3)^2 + (x_2 - 2x_3)^4$$

Again, let's calculate on BOARD!

Newton's Method: Solution

$$x^{(0)} = [3, -1, 0]$$

$$f(x_1, x_2, x_3) = (x_1 + 10x_2)^2 + 5(x_1 - x_3)^2 + (x_2 - 2x_3)^4$$

Newton's Method Example in one step:

- ➔ Calculate $\nabla f(x^{(n)})$
- ➔ Calculate $F(x^{(n)})$
- ➔ Calculate $[F(x^{(n)})]^{-1}$
- ➔ Update: $x^{(n+1)}$
- ➔ Update: $n = n + 1$

$$\nabla f(x^{(0)}) = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3}\right] = [16, -144, 22]$$

$$F(x^{(0)}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_1 \partial x_3} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \frac{\partial^2 f}{\partial x_2 \partial x_3} \\ \frac{\partial^2 f}{\partial x_3 \partial x_1} & \frac{\partial^2 f}{\partial x_3 \partial x_2} & \frac{\partial^2 f}{\partial x_3^2} \end{bmatrix} = \begin{bmatrix} 12 & 20 & -10 \\ 20 & 22 & -24 \\ -10 & -24 & 48 \end{bmatrix}$$

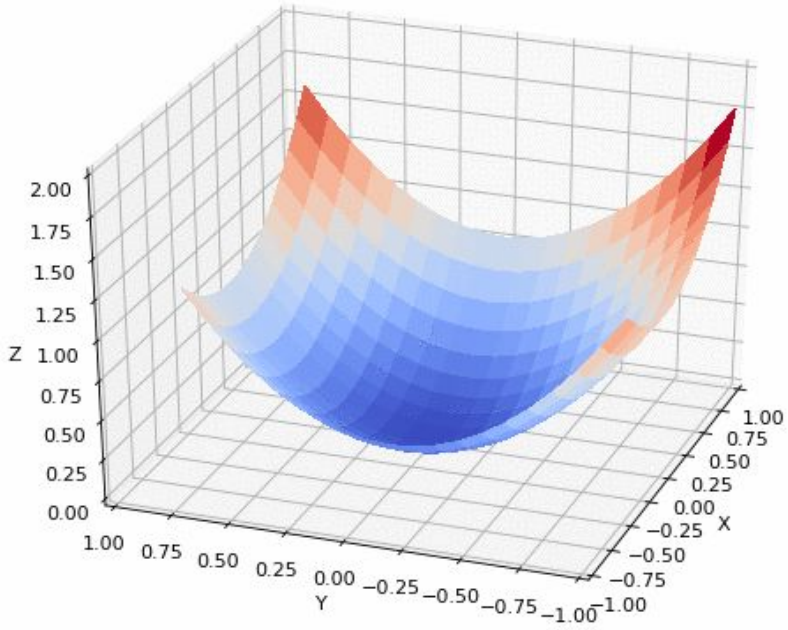
$$[F(x^{(0)})]^{-1} = \begin{bmatrix} -0.079 & 0.119 & 0.043 \\ 0.119 & -0.079 & -0.015 \\ 0.043 & -0.015 & 0.023 \end{bmatrix}$$

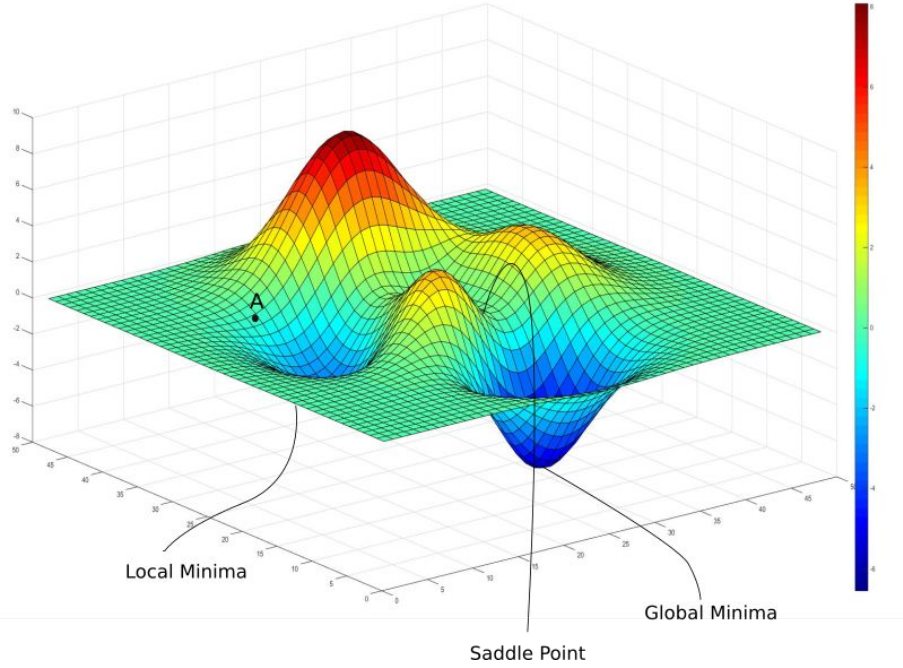
$$x^{(1)} = x^{(0)} - [F(x^{(0)})]^{-1} \cdot \nabla f(x^{(0)})$$

$$n = 1$$

UCLA

Gradient Descent

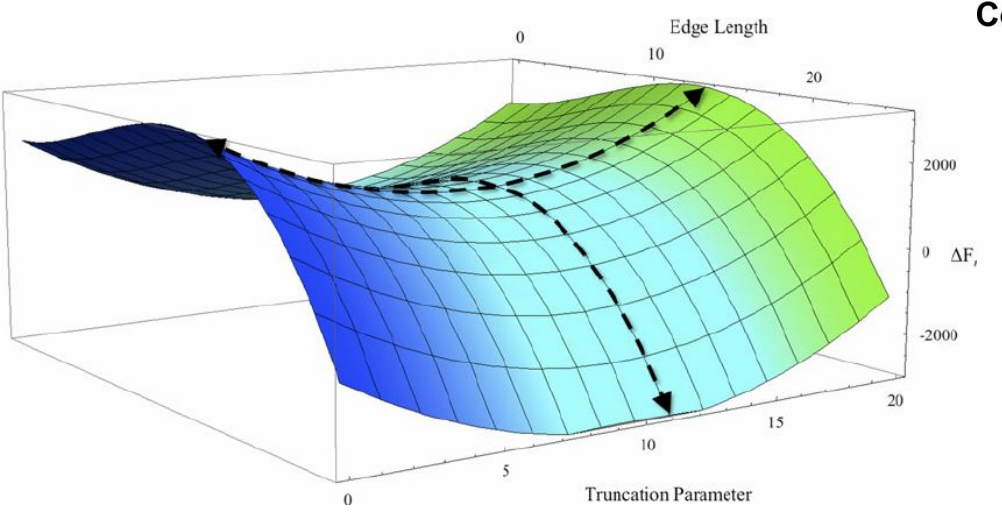


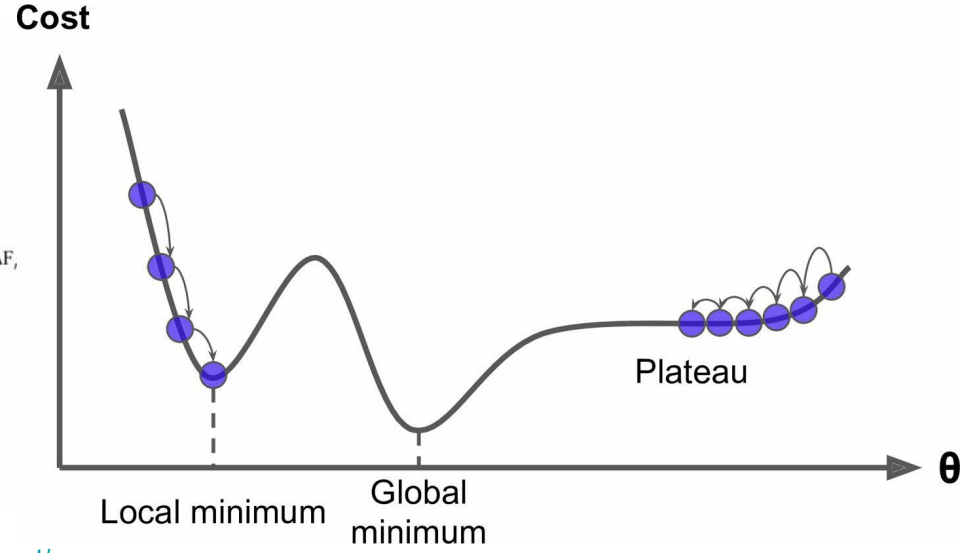


Gradient Descent: Batch vs Stochastic

Why do we need Stochastic GD besides the efficiency/scalability reason?

Think about saddle points and local minima.



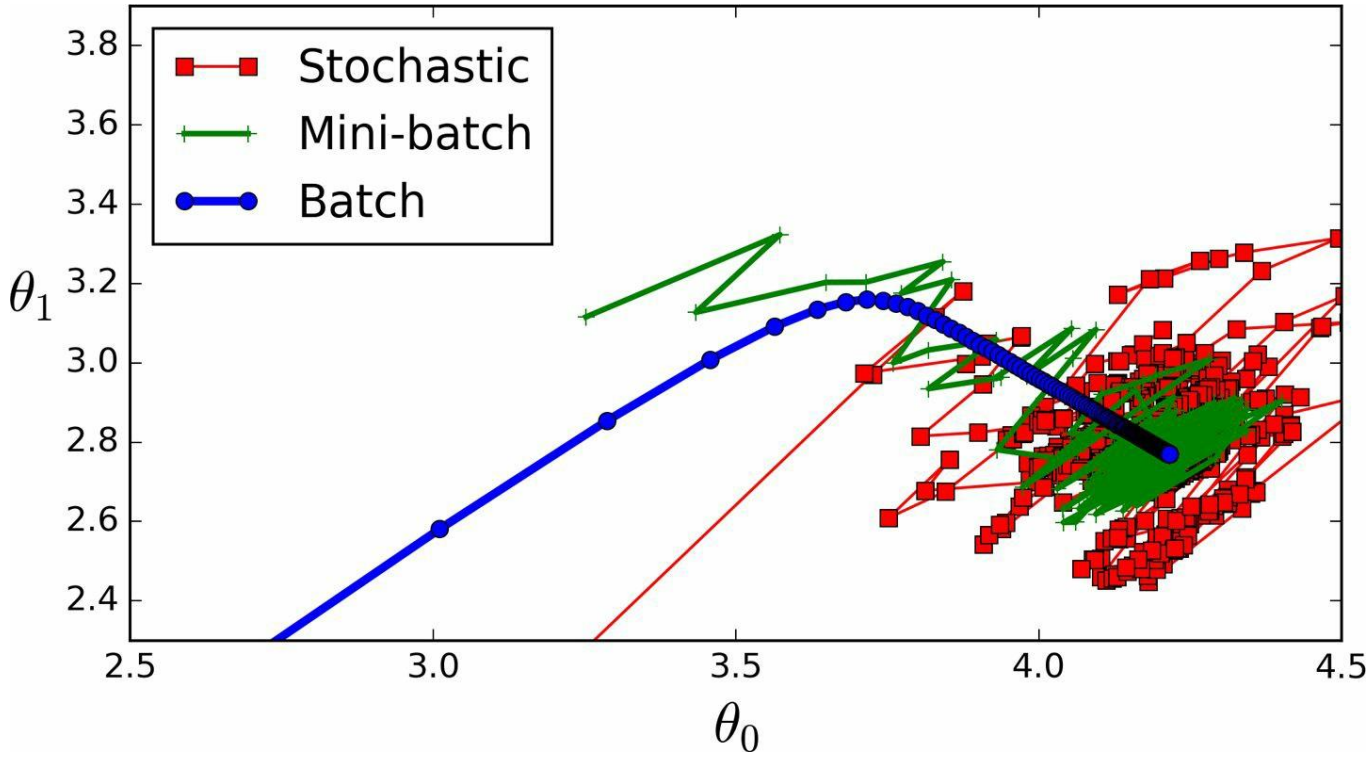


https://blog.paperspace.com/intro-to-optimization-in-deep-learning-gradient-descent/

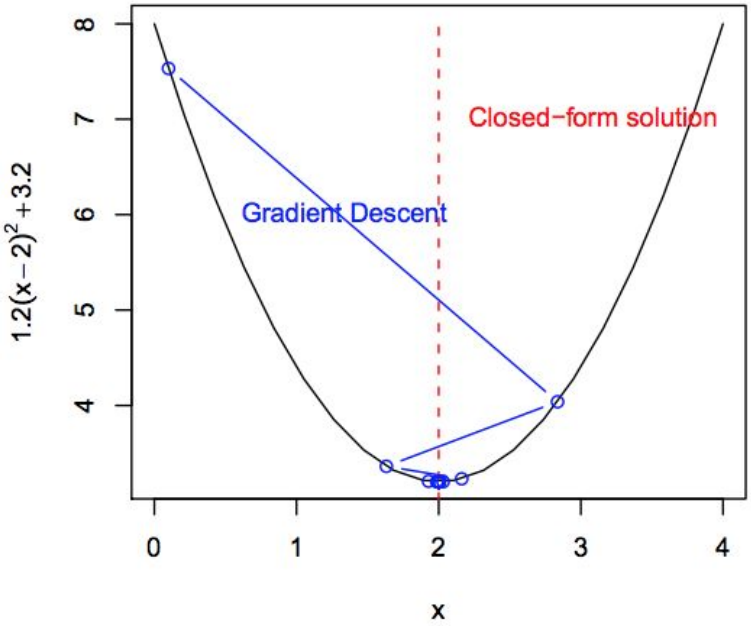
Géron, Aurélien. *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. " O'Reilly Media, Inc.", 2017.

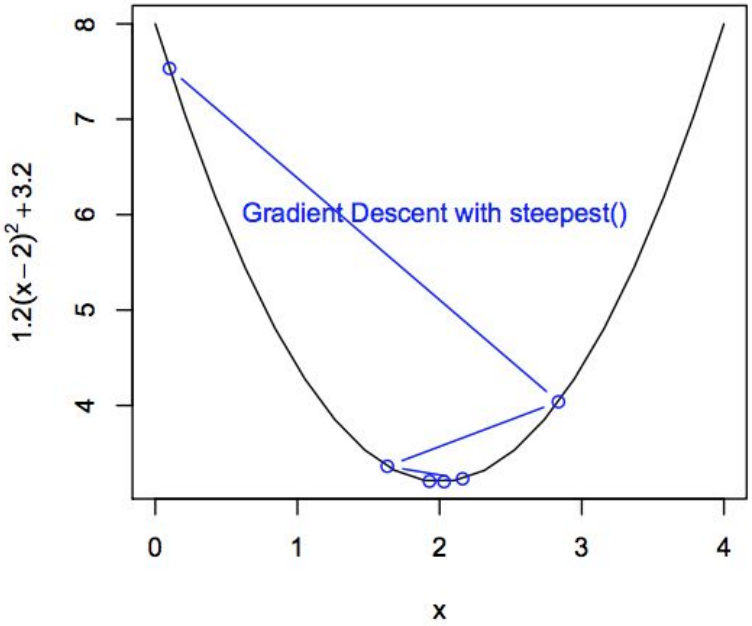
Comparison

Batch vs Stochastic vs Mini-batch



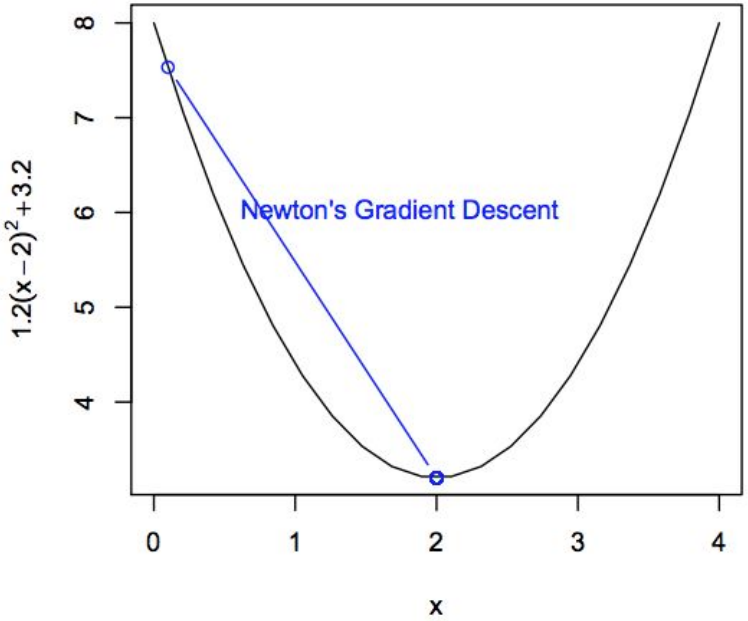
UCLA Gradient Descent or Newton's Method?

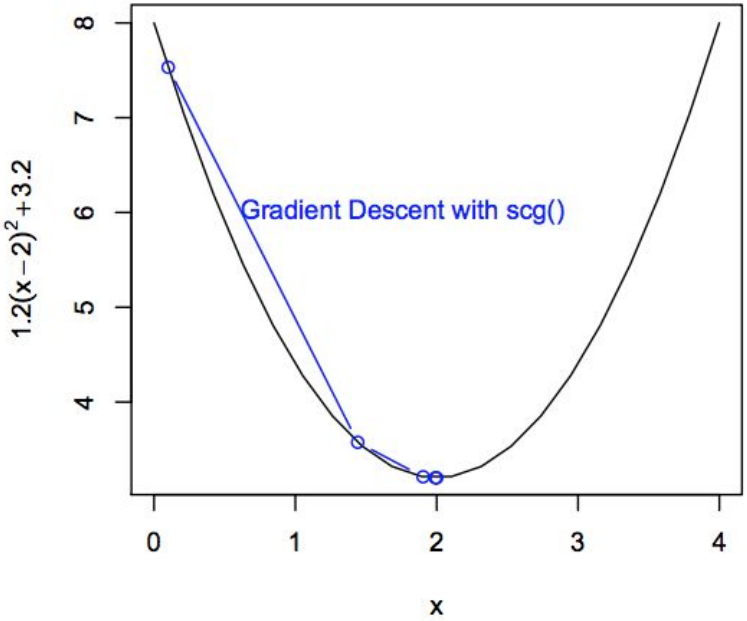




https://stats.stackexchange.com/questions/253632/why-is-newtons-method-not-widely-used-in-machine-learning
https://math.stackexchange.com/questions/609680/newtons-method-intuition
https://thelaziestprogrammer.com/sharrington/math-of-machine-learning/solving-logreg-newtons-method
https://web.archive.org/web/20151122203025/http://www.cs.colostate.edu/~anderson/cs545/Lectures/week6day2/week6day2.pdf

UCLA Gradient Descent or Newton's Method?





https://stats.stackexchange.com/questions/253632/why-is-newtons-method-not-widely-used-in-machine-learning
https://math.stackexchange.com/questions/609680/newtons-method-intuition
https://thelaziestprogrammer.com/sharrington/math-of-machine-learning/solving-logreg-newtons-method
https://web.archive.org/web/20151122203025/http://www.cs.colostate.edu/~anderson/cs545/Lectures/week6day2/week6day2.pdf

Logistic Regression: Example Question

We are given a data set consisting of the following experiment. Well, the dataset is a little bit small. (O_o)

The height and weight of 3 people were recorded at the beginning of each person's 65th birthday. At exactly one year after each person's 65th birthday the vital status was recorded to be either alive or deceased.

Our end goal is to use logistic regression to predict the probability that a person's life expectancy is at least 66 years given their age of 65, initial vital status of alive, height, and weight (but we won't go that far here).

The data is given in the following table on the right.

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

Logistic Regression: Example Question

Step 1: State the log-likelihood function.

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

Logistic Regression: Example Question

Step 1: State the log-likelihood function.

Answer:

$$\alpha_1 = -b - 155w_1 - 60w_2$$

$$\alpha_2 = -b - 135w_1 - 64w_2$$

$$\alpha_3 = -b - 170w_1 - 73w_2$$

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

$$L = log\left(1 - \frac{1}{1+e^{\alpha_1}}\right) + log\left(\frac{1}{1+e^{\alpha_2}}\right) + log\left(\frac{1}{1+e^{\alpha_3}}\right)$$

Logistic Regression: Example Question

Step 2: State the gradients for each parameter.

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

UCLA Logistic Regression: Example Question

Step 2: State the gradients for each parameter.

Answer:

$$L = log\left(1 - \frac{1}{1+e^{\alpha_1}}\right) + log\left(\frac{1}{1+e^{\alpha_2}}\right) + log\left(\frac{1}{1+e^{\alpha_3}}\right)$$

$$\nabla_b = -1.0 \cdot \frac{1}{1+e^{\alpha_1}} + -1.0 \cdot -\frac{e^{\alpha_2}}{1+e^{\alpha_2}} + -1.0 \cdot -\frac{e^{\alpha_3}}{1+e^{\alpha_3}}$$

$$\nabla_{w_1} = -155.0 \cdot \frac{1}{1+e^{\alpha_1}} + -135.0 \cdot -\frac{e^{\alpha_2}}{1+e^{\alpha_2}} + -170.0 \cdot -\frac{e^{\alpha_3}}{1+e^{\alpha_3}}$$

$$\nabla_{w_2} = -60 \cdot \frac{1}{1+e^{\alpha_1}} + -64.0 \cdot -\frac{e^{\alpha_2}}{1+e^{\alpha_2}} + -73.0 \cdot -\frac{e^{\alpha_3}}{1+e^{\alpha_3}}$$

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

$$\alpha_1 = -b - 155w_1 - 60w_2$$

$$\alpha_2 = -b - 135w_1 - 64w_2$$

$$\alpha_3 = -b - 170w_1 - 73w_2$$

Logistic Regression: Example Question

Step 3: Give the Hessian Matrix

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

Logistic Regression: Example Question

Step 3: Give the Hessian Matrix

$$H_b^T = \begin{bmatrix} -1.0 \cdot -1.0 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -1.0 \cdot -1.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -1.0 \cdot -1.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \\ -155.0 \cdot -1.0 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -135.0 \cdot -1.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -170.0 \cdot -1.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \\ -60 \cdot -1.0 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -64.0 \cdot -1.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -73.0 \cdot -1.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \end{bmatrix}$$

$$\nabla_b = -1.0 \cdot \frac{1}{1+e^{\alpha_1}} + -1.0 \cdot -\frac{e^{\alpha_2}}{1+e^{\alpha_2}} + -1.0 \cdot -\frac{e^{\alpha_3}}{1+e^{\alpha_3}}$$

$$\nabla_{w_1} = -155.0 \cdot \frac{1}{1+e^{\alpha_1}} + -135.0 \cdot -\frac{e^{\alpha_2}}{1+e^{\alpha_2}} + -170.0 \cdot -\frac{e^{\alpha_3}}{1+e^{\alpha_3}}$$

$$\nabla_{w_2} = -60 \cdot \frac{1}{1+e^{\alpha_1}} + -64.0 \cdot -\frac{e^{\alpha_2}}{1+e^{\alpha_2}} + -73.0 \cdot -\frac{e^{\alpha_3}}{1+e^{\alpha_3}}$$

$$H_{w_1}^T = \begin{bmatrix} -1.0 \cdot -155.0 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -1.0 \cdot -135.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -1.0 \cdot -170.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \\ -155.0 \cdot -155.0 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -135.0 \cdot -135.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -170.0 \cdot -170.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \\ -60 \cdot -155.0 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -64.0 \cdot -135.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -73.0 \cdot -170.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \end{bmatrix}$$

$$H_{w_2}^T = \begin{bmatrix} -1.0 \cdot -60 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -1.0 \cdot -64.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -1.0 \cdot -73.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \\ -155.0 \cdot -60 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -135.0 \cdot -64.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -170.0 \cdot -73.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \\ -60 \cdot -60 \cdot -\frac{e^{\alpha_1}}{(1+e^{\alpha_1})^2} + -64.0 \cdot -64.0 \cdot -\frac{e^{\alpha_2}}{(1+e^{\alpha_2})^2} + -73.0 \cdot -73.0 \cdot -\frac{e^{\alpha_3}}{(1+e^{\alpha_3})^2} \end{bmatrix}$$

UCLA Logistic Regression: Example Question

Step 4: Assuming an initial guess of 0.25 for each parameter, write python code for finding the values of the parameters after 2 iterations using the Newton Raphson method.

$$b = 1.1346728128592689$$

$$w_1 = -2.4878423877892759$$

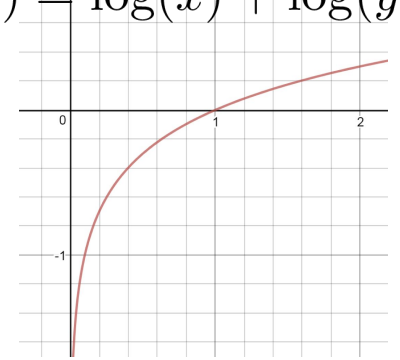
$$w_2 = 3.8192554544178936$$

Height (inches)	Weight (lbs)	Vital Status
60	155	Deceased
64	135	Alive
73	170	Alive

Notes on Log-likelihood

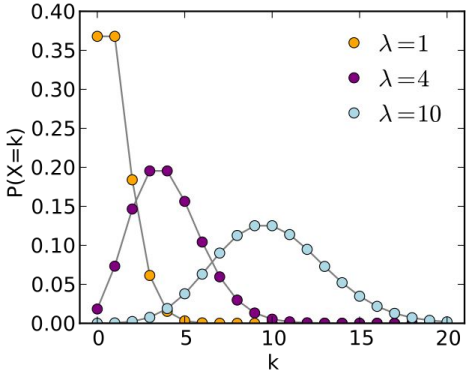
Logistic Regression

$$\log(x \cdot y) = \log(x) + \log(y)$$



- Is likelihood a density or probability?
  - No, it is the multiplication of densities
- Log-likelihood used in gradient method
  - Logarithm is a strictly increasing function
  - Avoid product of probabilities underflow
  - Easier to take partial derivative than multiplication of probabilities
- Examples → Maximum Likelihood on Poisson Distribution

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

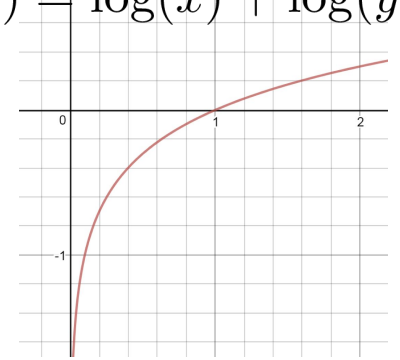


UCLA

Notes on Log-likelihood

Logistic Regression

$$\log(x \cdot y) = \log(x) + \log(y)$$



- Is likelihood a density or probability?
  - No, it is the multiplication of densities
- Log-likelihood used in gradient method
  - Logarithm is a strictly increasing function
  - Avoid product of probabilities underflow
  - Easier to take partial derivative than multiplication of probabilities
- Examples → Maximum Likelihood on Poisson Distribution

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

$$\begin{aligned} l(\lambda) &= \sum_{i=1}^{n} (X_i log\lambda - \lambda - logX_i!) \\ &= log\lambda \sum_{i=1}^{n} X_i - n\lambda - \sum_{i=1}^{n} logX_i! \end{aligned}$$

$$l'(\lambda) = \frac{1}{\lambda} \sum_{i=1}^{n} x_i - n = 0 \longrightarrow \hat{\lambda} = \bar{X}$$

UCLA

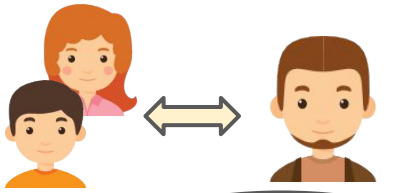
Project: Help Yelp!

Course project contest on yelp rating prediction is launched on Kaggle.

- Link: https://www.kaggle.com/c/yelpratingprediction
- Important Dates
  - **Today: Group Formation**
  - Nov. 5: Midterm Report
  - Dec. 9: Final Report

UCLA

Yelp Dataset



Friends

User

Review

Business

User ID

Friends

Average stars

Personal Features

Account Duration

Review counts

User ID

Business ID

Review Stars

Review Text

Date

Useful

Business ID

Location

Attributes

Categories

Average Stars

Hours

Business

- Index
- Features / Attributes (outdoor/is_romantic, etc)
- Categories
- Hours
- Address (Street/Postal code/state)
- Stars

User

- Index
- Features
  - Fans
  - Is_cool
  - Useful score
  - etc
- Review counts
- Friends User ID
- Average stars


```
{u'average_stars': 4.21,
 u'compliment_cool': 1,
 u'compliment_cute': 0,
 u'compliment_funny': 1,
 u'compliment_hot': 0,
 u'compliment_list': 0,
 u'compliment_more': 0,
 u'compliment_note': 0,
 u'compliment_photos': 0,
 u'compliment_plain': 1,
 u'compliment_profile': 0,
 u'compliment_writer': 1,
 u'cool': 2,
 u'elite': u'None',
 u'fans': 3,
 u'friends': u'ea3HFxRRo4gZOmlDle8IPA, UXpTrzITc0yg08sXftsa-w, 6SO8-96FGuvTEZjXE1lK9A, LKe5yggs-alTn1BnEWxcvA,
wskTNZjwiqMJOvM_-XAQZw, -3yzjh7rkfNr6_m0J80hFg, 9BGpVdzZ0FxuytgA9eOwqA, dhPp4t9_fD0R0eNsp_gYZw, wQHvfghJ9V1gZJC
4xOLQfw, bc-ZRzH15AN5m2EmNPrmRQ, d6TG842sXMiH_cExQ22zgA, N-0uq1uev8y-AaCAPrnJtw, BCb9inKZcyExxaAxQoD2HA, 5j65np
tHTM7E5G-Lclyz8Q, MKDVsEzQK48og5S0MItoyg, 1xhs7kJ_ZeiPcMP_bJpBJQ, ZMwJaTjanot_jVy35RNSAg, HfveiyvqjnU2scNrLvmkb
A, a0Njyhat4a2I-88G6fit8A, f60urWTVLrGlCH4ZH0MlWw, quC87zSHWK4CtThl4fdurg, 5Kk7CSyCKiOyUx5dSkjHWw, Bpj4zZzHAo7K
ZWLNJmeOjA, mKemb3MpSyUbsVUVSmI5tg, RSNTkVCsjN44WBlXK5aHwg, OiLvXTJIdH1rN3exnVvWSw, ZrN6jjXI41GP7Oieae6CNQ, QKM
mhTjCMBBnOVFxY4Jitg, IXqSUuzukAkpF7yY9MkiZw',
 u'funny': 6,
 u'name': u'Christine',
 u'review_count': 34,
 u'useful': 22,
 u'user_id': u'jhpYtJj6mijxNGWRARt7ew',
 u'yelping_since': u'2010-08-09'}
```

Review - Links between user and item

- Text
- Stars
- User ID
- Restaurant ID


```
{u'business_id': u'pomGBqfbxcqPv14c3XH-ZQ',
 u'cool': 0,
 u'date': u'2012-11-13',
 u'funny': 0,
 u'review_id': u'dDl8zu1vWPdKGihJrwQbpw',
 u'stars': 5,
 u'text': u'I love this place! My fiance And I go here atleast once a week. The portions are huge! Food is amaz
ing. I love their carne asada. They have great lunch specials... Leticia is super nice and cares about what you
 think of her restaurant. You have to try their cheese enchiladas too the sauce is different And amazing!!!',
 u'useful': 0,
 u'user_id': u'msQe1u7Z_XuqjGoqhB0J5g'}
```

```
{u'business_id': u'f-v1fvtnbdw_QQRsCnwH-g',
 u'cool': 0,
 u'date': u'2017-11-18',
 u'funny': 0,
 u'review_id': u'alI_kRKyEHfdHibYGgtJbw',
 u'stars': 1,
 u'text': u'I have to write a review on the Fractured Prune. First off I love doughnuts second I used to work i
n a bakery. So I know a little about the almighty doughnut. I was there this past Thursday night at Westgate wh
ere I was having a nice dinner at Opah Life. I wanted a little desert so I went to this doughnut shop it seemed
 clean enough so I made my selections and took them home. Mistake should have eaten one there I would have ask
for my money back. I took a bite out of each on to confirm what I thought they were the worst doughnuts I have
ever had. Horrible not sure if the oils was rancid or not hot enough either way eat at your own risk.',
 u'useful': 0,
 u'user_id': u'Fc_nb6N6Sdurqb-rwsY1Bw'}
```

UCLA

Evaluation: RMSE

Evaluation metrics: RMSE (Root-mean-square error) between ground truth ratings and predicted ratings between user-business pair.

Formula:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

UCLA

Some explanations

- Data is in various types including **text data, binary features, numerical features, categorical features, etc.**
- If you consider friendship among users, it naturally constructs a *“user graph”*.
- If you consider business in same area, it also forms another *“business map”* based on geographic relations.
- There are temporal-related features on reviews.
- Some features of users and business are missing.
  - For example, some users do not have friends.
  - Some business attributes, hours or neighborhood are unknown.
- **Not all information are useful! → Some features may not be helpful for rating tasks.**
- **Try to start simple models. Sometimes they surprisingly work.**

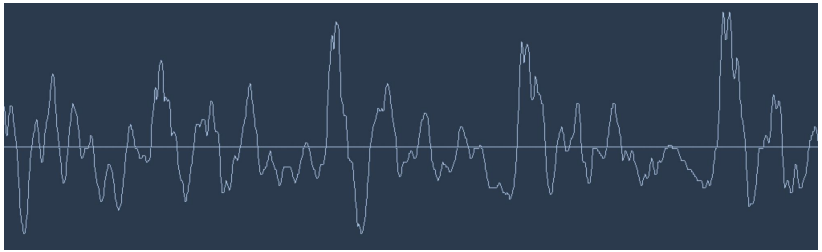
UCLA

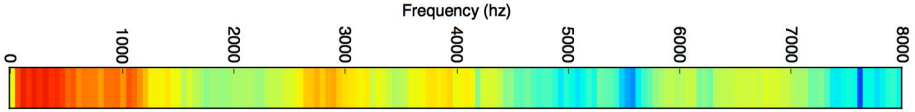
Feature Engineering

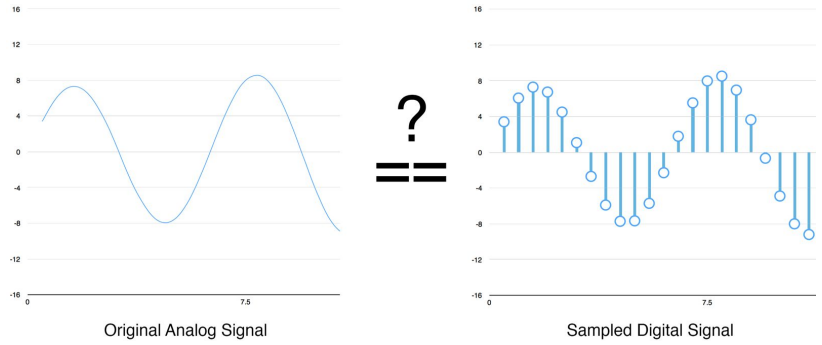
- What is feature engineering?
  - Wikipedia: The process of using **domain knowledge** of the data to **create features** that make machine learning algorithms work.
- Reason for feature engineering
  - Simpler models / Better results
- When to apply feature engineering
  - Data collection → Data cleaning → **Feature Engineering** → Model Training → Evaluation
- Examples
  - Speech recognition: Use Fourier transform to analyze spectrum instead of time series
  - Many other applications …
- Methods: **Selection / Construction / Representation**
  - Example: Indicator variables
  - More feature engineering methods
- Advanced topic: Automated feature engineering

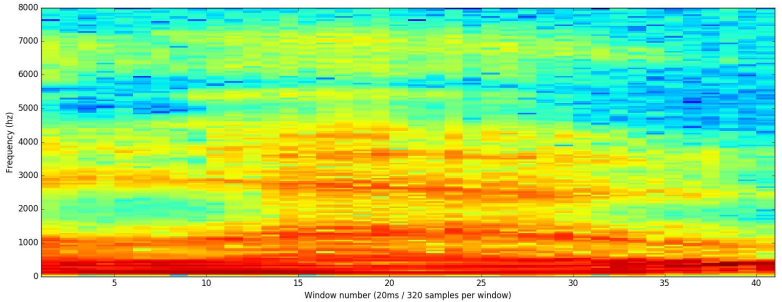
Speech Recognition

Examples of Feature Engineering









Check Blog: https://medium.com/@ageitgey/machine-learning-is-fun-part-6-how-to-do-speech-recognition-with-deep-learning-28293c162f7a

UCLA Programming Tips: `Numpy/Pandas`

- Numpy: A fundamental package for scientific computing with Python. It has following functions
  - Operations of N-dimensional array object (matrix, tensor)
  - Useful linear algebra, mathematical transform, and random number capabilities
- Pandas: Provide high-performance, easy-to-use data structures and data analysis toolkit for the Python language
- Installation Guide: → Install `numpy` → Install `pandas`
- Tutorials: → `numpy` quickstart → `pandas` tutorial
- Cheatsheet: → `numpy` cheatsheet → `pandas` cheatsheet

UCLA Programming Tips: `Numpy/Pandas`

- What do you need from `numpy` and `pandas` in homework?

`Numpy:`

(1) How to create/initialize a numpy array/matrix object
(2) Dimensions of numpy array
(3) Basic operations of numpy array (add, subtract, product, etc)
(4) Indexing, slicing and stacking different arrays
(5) Broadcasting rule

`Pandas:`

(1) How to load data from file
(2) How data is stored in DataFrame?
(3) How to read and write data by index in DataFrame?
(4) Iterating over a pandas DataFrame

UCLA

Thank you!

Enjoy "mining" and good luck!