# Facial Analyzer for Rhinoplasty (Option 9)

Priscilla Carbo
Marvin Cazeau
Jared Curtis
Maree Kelly
Alessandra Oo

# Table of Contents

# Last Semester

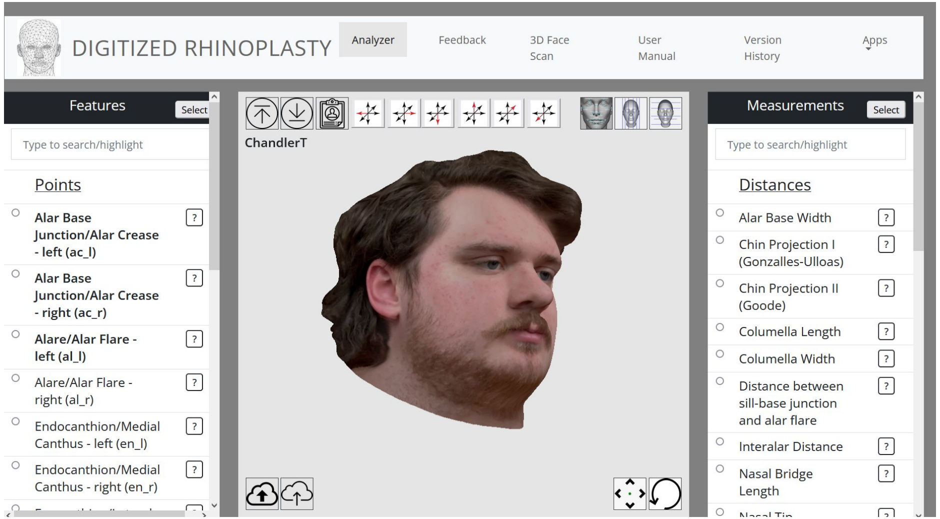
- Suggesting changes to patient’s nose
  - Based on similarly structured faces
- Create clusters of patients
  - Based on K-Nearest-Neighbors algorithm
- Fit patients to clusters
  - Based on appearance using a Convolutional Neural Network
- Adding to our database using scanned faces
  - Implementing Bellus3D software

# Goals for this Semester

- Create a consistent method of synthetic face creation to fill database
    - This would provide better facial groupings as it has more faces to compare and analyze
- Generate more accurate facial clusters
    - This code would take all the faces from the database and group them based on how similar they are to each other as well as other variables
- Plotting facial landmarks automatically

# Database: Scanned faces

- 96 scanned faces using Bellus3D (iOS application, discontinued)
- 114 3D faces generated by FaceGen Modeller with 2D images
- Together we have 210 faces

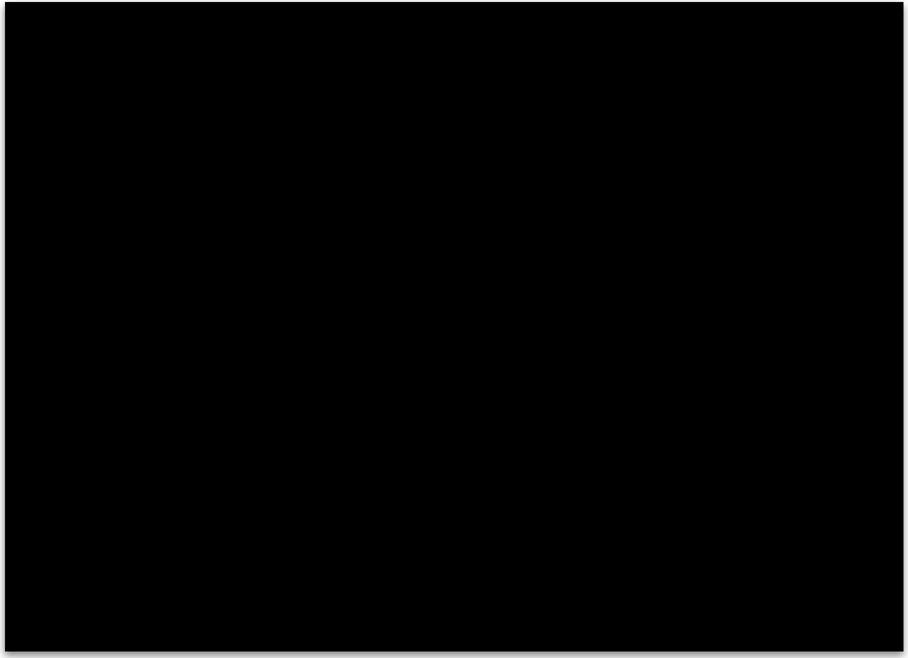


# FaceGen

- Main purpose for this software was to grow our database
  - We previously used Bellus3D app for this

- Researched various machine learning algorithms to create 3D models for dataset
  - GANs?

- Constraints with FaceGen:
  - A 3 JPGs are required
    - 2 profile and 1 front face



# Database: Digitized Rhinoplasty

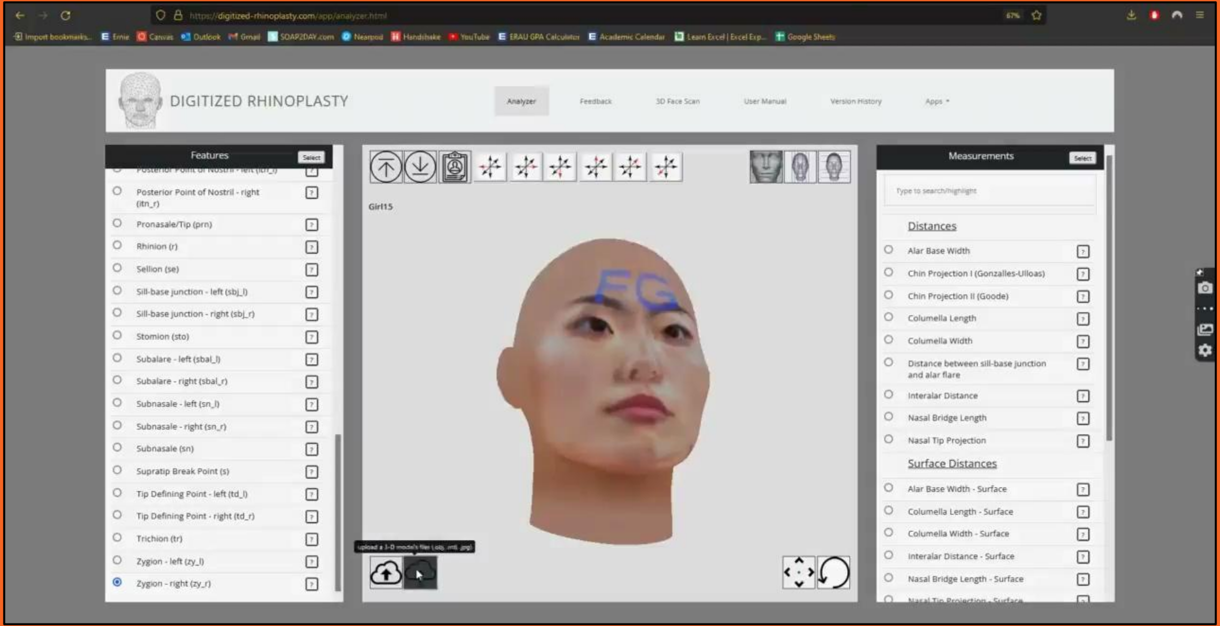
- We used Digitized Rhinoplasty web application to point the landmarks manually on each 3D faces.

- We are using 45 of the most significant landmarks.
  - 14 of the most commonly occurring landmarks in studies that have the highest relation to facial structure.
  - 31 landmarks that affect the nose structure.

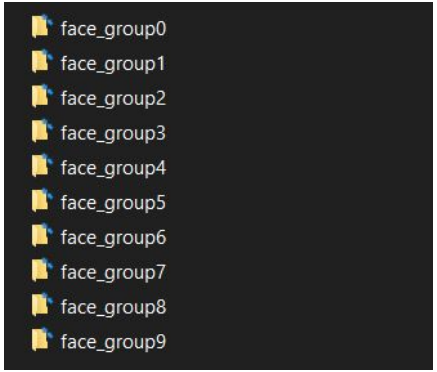
| Facial Landmark | Frequency in Important Measurements | Frequency of Studies in Literature |
|---|---|---|
| Alar Base Junction/Alar Crease - left/right (ac) | 3 | 9 |
| Alar Rim's Highest Point - left/right (armax) | 1 | 0 |
| Alare/Alar Flare - left/right (al) | 1 | 8 |
| Cheilion - left/right (ch) | 0 | 19 |
| Columellar Break Point (cb) | 4 | 0 |
| Columellar Rim - left/right (cmin) | 1 | 0 |
| Crista Philtri - left/right (cph) | 0 | 4 |
| Endocanthion/Medial Canthus - left/right (en) | 2 | 15 |
| Exocanthion/Lateral Canthus - left/right (ex) | 1 | 22 |
| Glabella (g) | 4 | 2 |
| Labiale Inferius (li) | 0 | 12 |
| Labiale Superius (ls) | 3 | 12 |
| Lateral helix of ear - left/right (la) | 1 | 0 |
| Maxillofrontale - left/right (ma) | 1 | 0 |
| Menton/Gnathion (me) | 2 | 4 |
| Nasal Parenthesis - left/right (np) | 1 | 0 |
| Nasion/Radix (n) | 6 | 15 |
| Pogonion (pg) | 0 | 6 |
| Pronasale/Tip (prn) | 9 | 24 |
| Pupil - left/right (p) | 0 | 4 |
| Stomion (sto) | 1 | 7 |
| Subalare - left/right (sbal) | 1 | 3 |
| Sublabiale/Mentolabial Sulcus (sl) | 0 | 4 |
| Subnasale (sn), Subnasale - left/right | 7 | 13 |
| Supratip Break Point (s) | 1 | 0 |
| Tip Defining Point - left/right (td) | 2 | 0 |
| Tragion - left/right (t) | 0 | 1 |
| Trichion (tr) | 2 | 0 |
| Zygion - left/right (zy) | 1 | 1 |

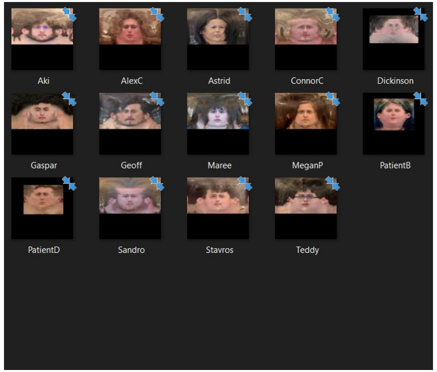
Table 1: Facial landmarks usage frequency in measurements important for rhinoplasty and facial landmarks studied in the literature for detection based on their geometric properties on 3D face scans.

# Digitized Rhinoplasty Web Application Demo



# Clustering

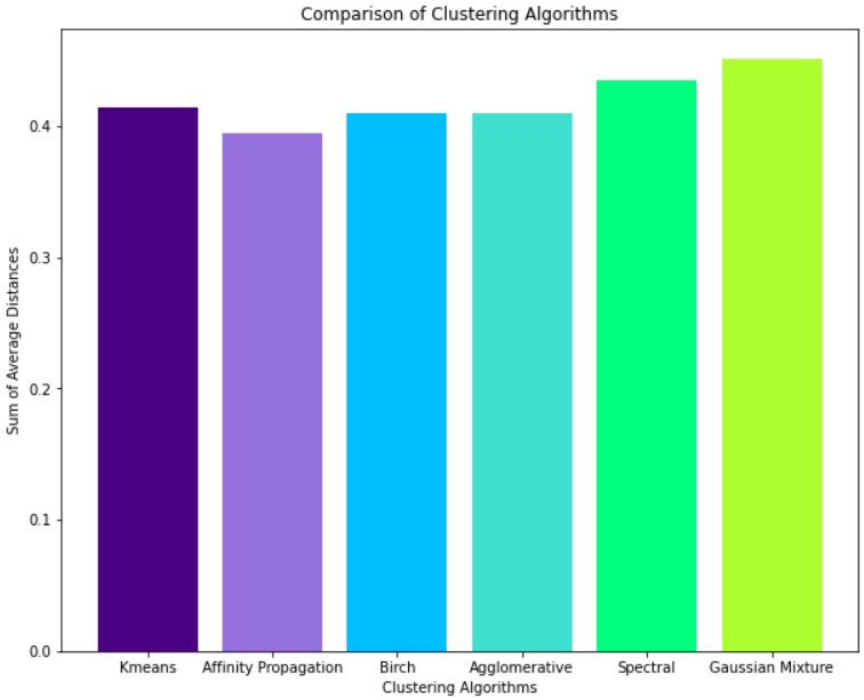


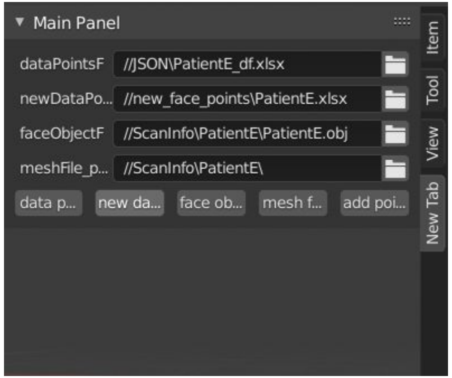

- Goal is to provide a patient with other faces that share similar facial structures
- Previously used a K nearest neighbors algorithm
  - However, we wanted fewer, more optimized groupings
- Provide a patient with multiple different clusters they can be fit into
  - Adds more options and variety of faces to choose from

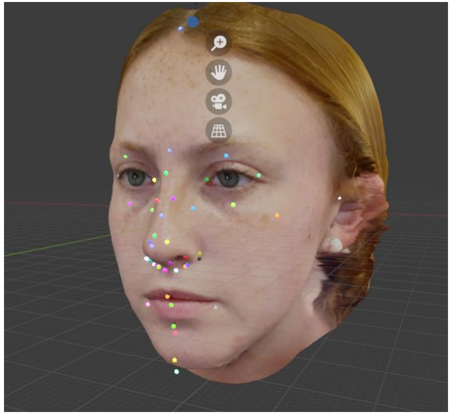
# Clustering

- 6 new clustering algorithms:
    - Kmeans
    - Affinity Propagation
    - BIRCH Clustering
    - Agglomerative Clustering
    - Spectral Clustering
    - Gaussian Mixture Model



# Blender Modeling

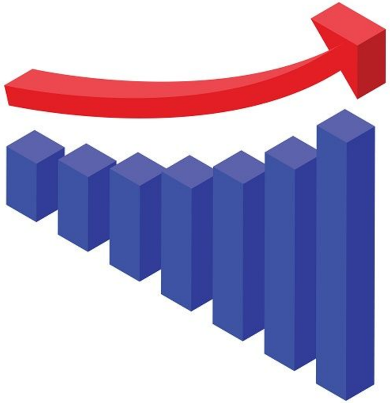


- Upload 3D object of patient's face
- Plot patient's original facial landmarks
- Plot patient's altered facial landmarks
  - Generated based on similarly structured faces
- Allow a user to sculpt the patient's 3D face
  - Make changes directly to the 3D model in order to provide a more optimally structured nose

# Blender Demo

# Room for Improvements

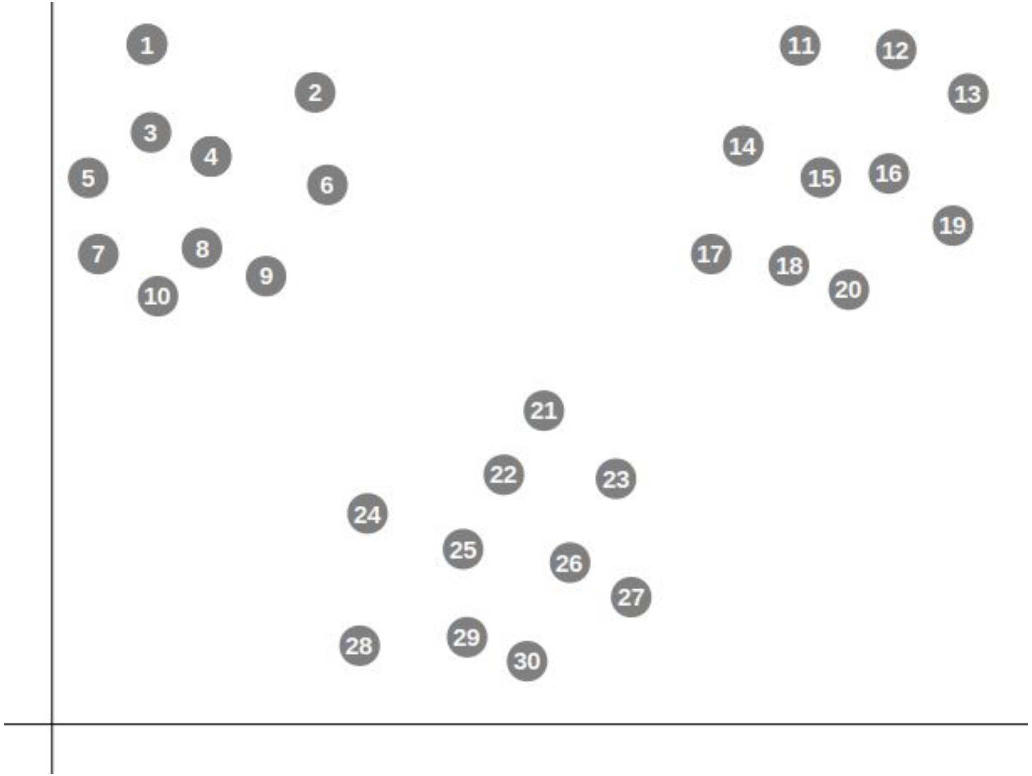


- Generate new nose landmarks for patient
  - Based on patients with similar facial structures
- Create an easy to use interface
  - Consolidate multiple python files into one coherent program
- Plot landmarks automatically
  - Use ML learning to plot facial landmarks
- Write code to generate 3D scan of patient’s face

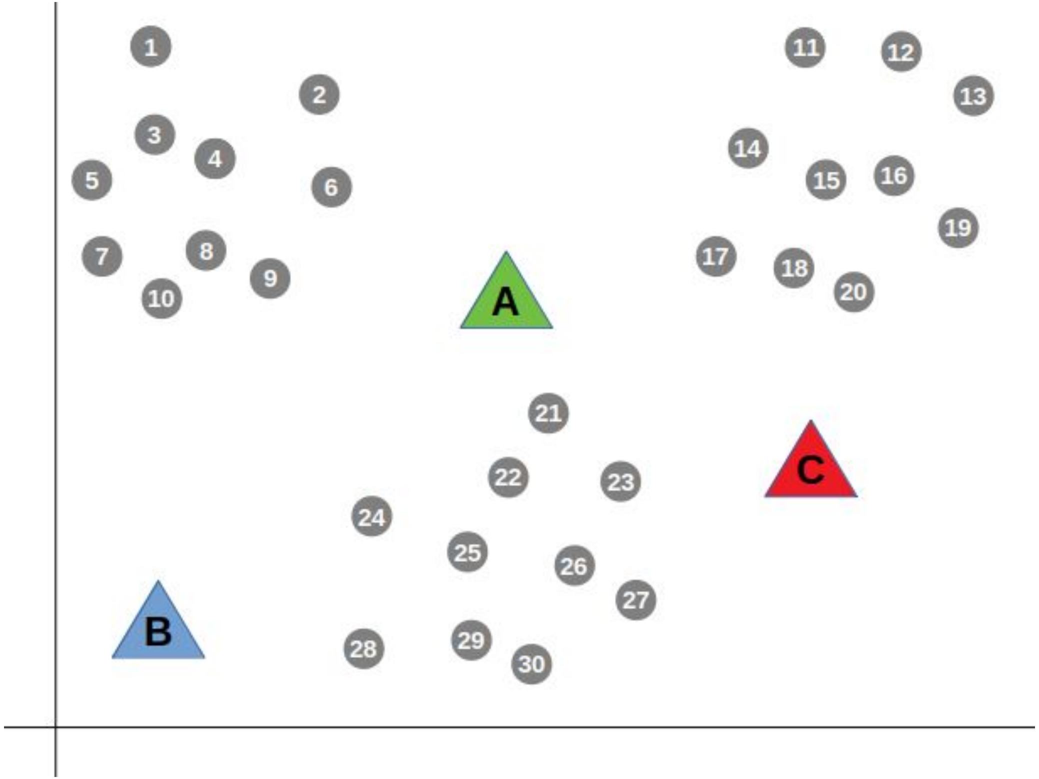
# Questions?

# Thank you.

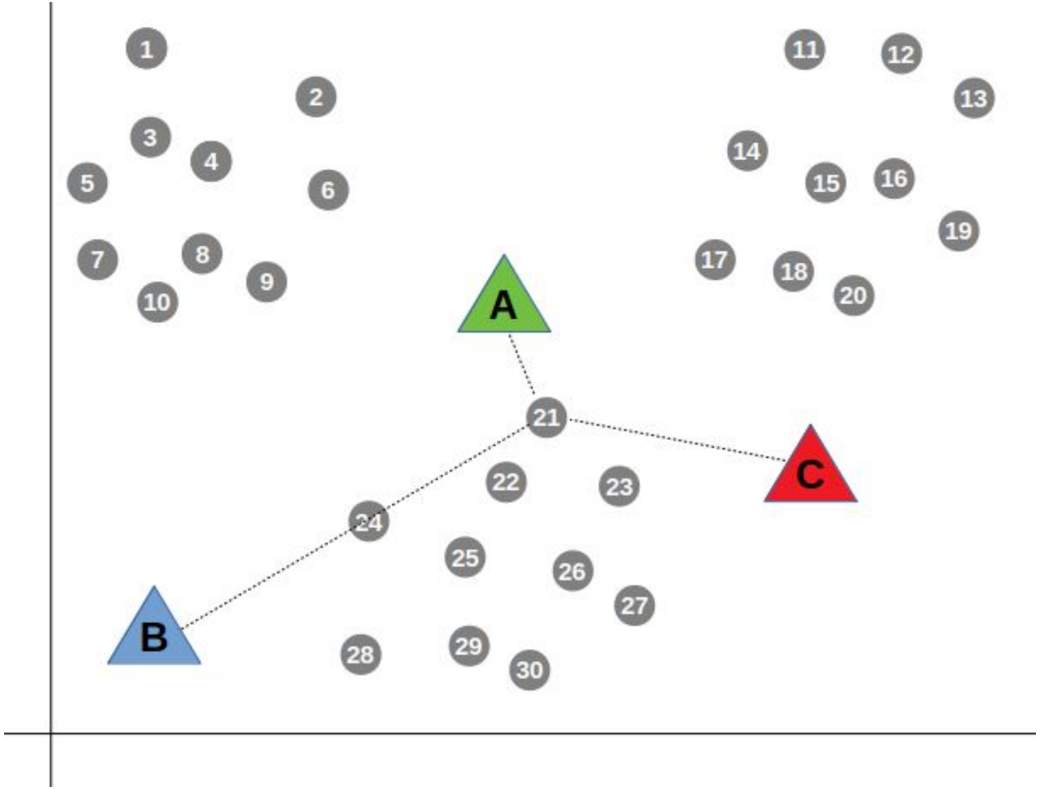
# Kmeans Clustering



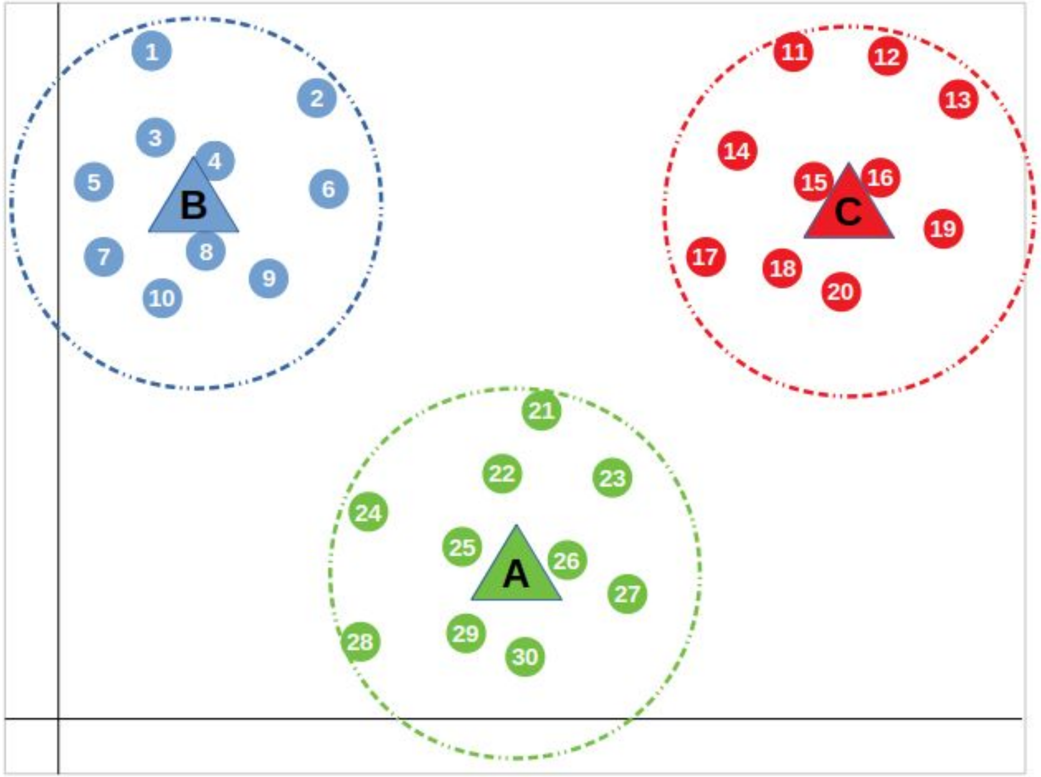
## Kmeans Clustering



# Kmeans Clustering



# Kmeans Clustering



## Affinity Propagation

**Table 1: Preferences of Five Participants**

| Participant | Tax Rate | Fee | Interest Rate | Quantity Limit | Price Limit |
|---|---|---|---|---|---|
| Alice | 3 | 4 | 3 | 2 | 1 |
| Bob | 4 | 3 | 5 | 1 | 1 |
| Cary | 3 | 5 | 3 | 3 | 3 |
| Doug | 2 | 1 | 3 | 3 | 2 |
| Edna | 1 | 1 | 3 | 2 | 3 |

## Affinity Propagation

### Table 2: Similarity Matrix

| Participant | Alice | Bob | Cary | Doug | Edna |
|---|---|---|---|---|---|
| Alice | -22 | -7 | -6 | -12 | -17 |
| Bob | -7 | -22 | -17 | -17 | -22 |
| Cary | -6 | -17 | -22 | -18 | -21 |
| Doug | -12 | -17 | -18 | -22 | -3 |
| Edna | -17 | -22 | -21 | -3 | -22 |

## Affinity Propagation

**Table 3: Responsibility Matrix**

| Participant | Alice | Bob | Cary | Doug | Edna |
|---|---|---|---|---|---|
| Alice | -16 | -1 | 1 | -6 | -11 |
| Bob | 10 | -15 | -10 | -10 | -15 |
| Cary | 11 | -11 | -16 | -12 | -15 |
| Doug | -9 | -14 | -15 | -19 | 9 |
| Edna | -14 | -19 | -18 | 14 | -19 |

## Affinity Propagation

### Table 4: Availability Matrix

| Participant | Alice | Bob | Cary | Doug | Edna |
|---|---|---|---|---|---|
| Alice | 21 | -15 | -16 | -5 | -10 |
| Bob | -5 | 0 | -15 | -5 | -10 |
| Cary | -6 | -15 | 1 | -5 | -10 |
| Doug | 0 | -15 | -15 | 14 | -19 |
| Edna | 0 | -15 | -15 | -19 | 9 |

# Affinity Propagation

## Table 5: Criterion Matrix

| Participant | Alice | Bob | Cary | Doug | Edna |
|---|---|---|---|---|---|
| Alice | **5** | -16 | -15 | -11 | -21 |
| Bob | **5** | -15 | -25 | -15 | -25 |
| Cary | **5** | -26 | -15 | -17 | -25 |
| Doug | -9 | -29 | -30 | **-5** | -10 |
| Edna | -14 | -34 | -33 | **-5** | -10 |

## Gaussian Mixture Model

- Begin with random Gaussian parameters for each Gaussian
  - Mean
  - Covariance
  - Weight
- Repeat expectation and maximization steps until convergence
  - Expectation Step
    - Compute p(zi = k | xi, θ). In other words, does sample *i* look like it came from cluster k?
  - Maximization Step
    - Update the Gaussian parameters (θ) to fit points assigned to them.

# Gaussian Mixture Model