949-734 / 90-835:
Designing Smart and Healthy Systems

Session 4: Learning for Patient Engagement

David Steier

November 2, 2021

Agenda

- HW#2
- Clustering
- Clustering to segment diabetic patients by engagement
- Prediction
- Predicting likelihood of diabetic patient readmission

HW #2

- All hypotheses, outcome and predictor variables collected from HW #1
- Each group uses the outcome and predictor variables to test hypotheses produced by another group
- Each group analyzes engagement patterns using k-medoids clustering
- Each group develops two models to predict patient engagement patterns.

Clustering

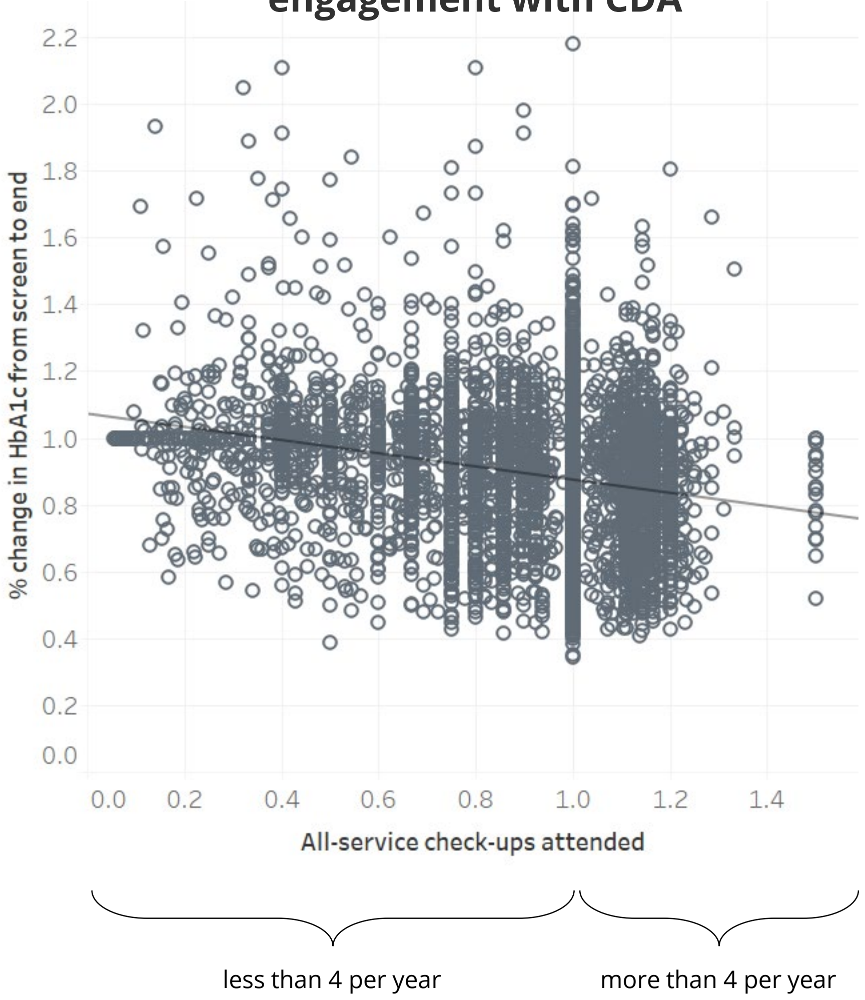
- The process of finding a “natural” partition of a dataset based on a set of variables
    - Each variable corresponds to a dimension of the data-space that will be partitioned
    - “Distance” in this data space is interpreted as “similarity”
    - Clusters are derived in such a way that items in one cluster are similar to one another; dissimilar from items in other clusters.
- There is no single notion of a cluster - there are many clustering algorithms
- Clustering is usually an iterative, judgment-intensive activity

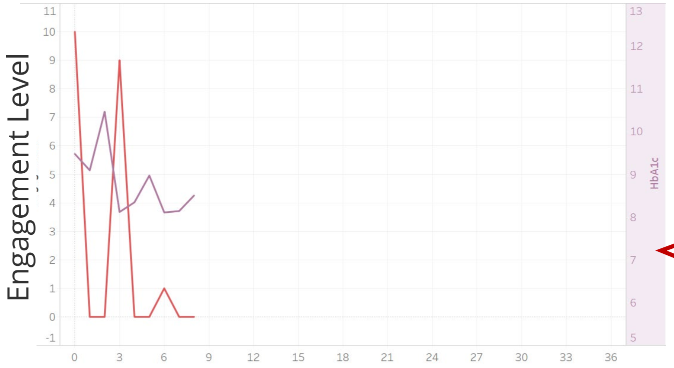
Slide credit: Deloitte Consulting LLP

Clustering to Understand Engagement Patterns

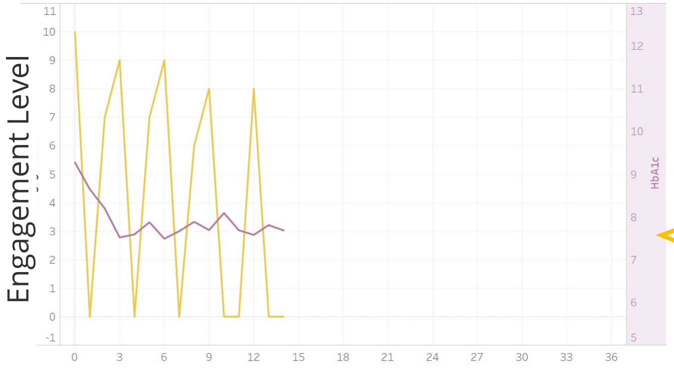
CDA's services are effective, and can be made even more effective - increasing engagement will likely improve health outcomes for patients across clinics

Patient reduction in HbA1c by engagement with CDA

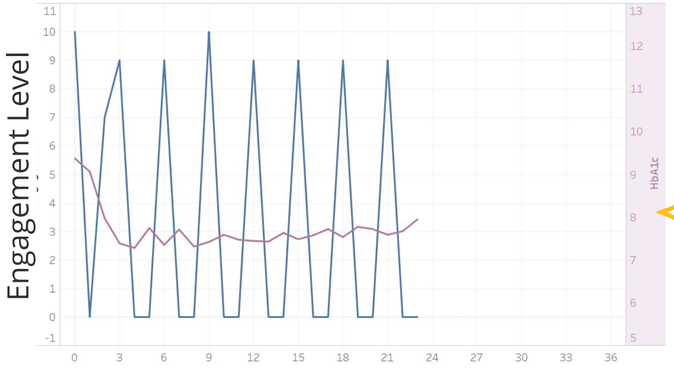




Low-engagement patients attend 1/3 of scheduled appointments and see a 1.3-point **(14%) reduction in HbA1c** in the first 6 months of treatment, from 9.4 to 8.2



Moderate-engagement patients attend 2/3 of scheduled appointments and see a 1.7-point **(18%) reduction in HbA1c** in the first 6 months of treatment, from 9.3 to 7.6



High-engagement patients attend 3/4 of scheduled appointments and see a 2-point **(22%) reduction in HbA1c** in the first 6 months of treatment, from 9.3 to 7.3

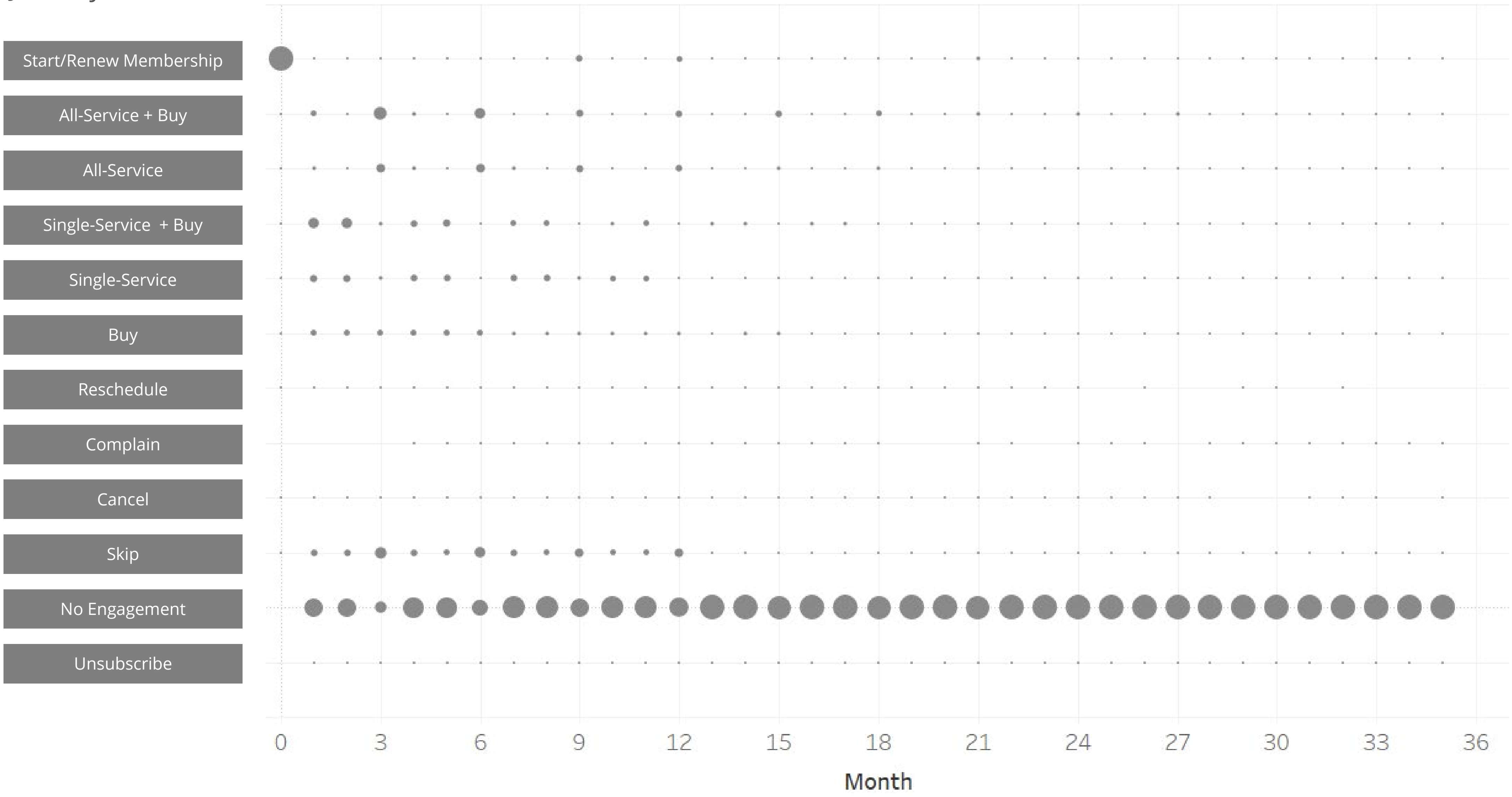
Events Are Signs of Engagement Levels

1. Cancels Membership
2. No Engagement
3. Skips Appointment
4. Cancels Appointment
5. Complains
6. Reschedules Appointment
7. Buys from Clinic
8. Attends Single-Service Appointment
9. Attends Single-Service Appointment and Buys from Clinic
10. Attends All-Service Appointment
11. Attends All-Service Appointment and Buys from Clinic
12. Buys or Renews (Annual) Membership

Slide credit: Deloitte Consulting LLP

Engagement Levels for CdA patients Over Time

The size of each circle corresponds to the proportion of patients who were at the given care level in the given month between January 13 and November 2016

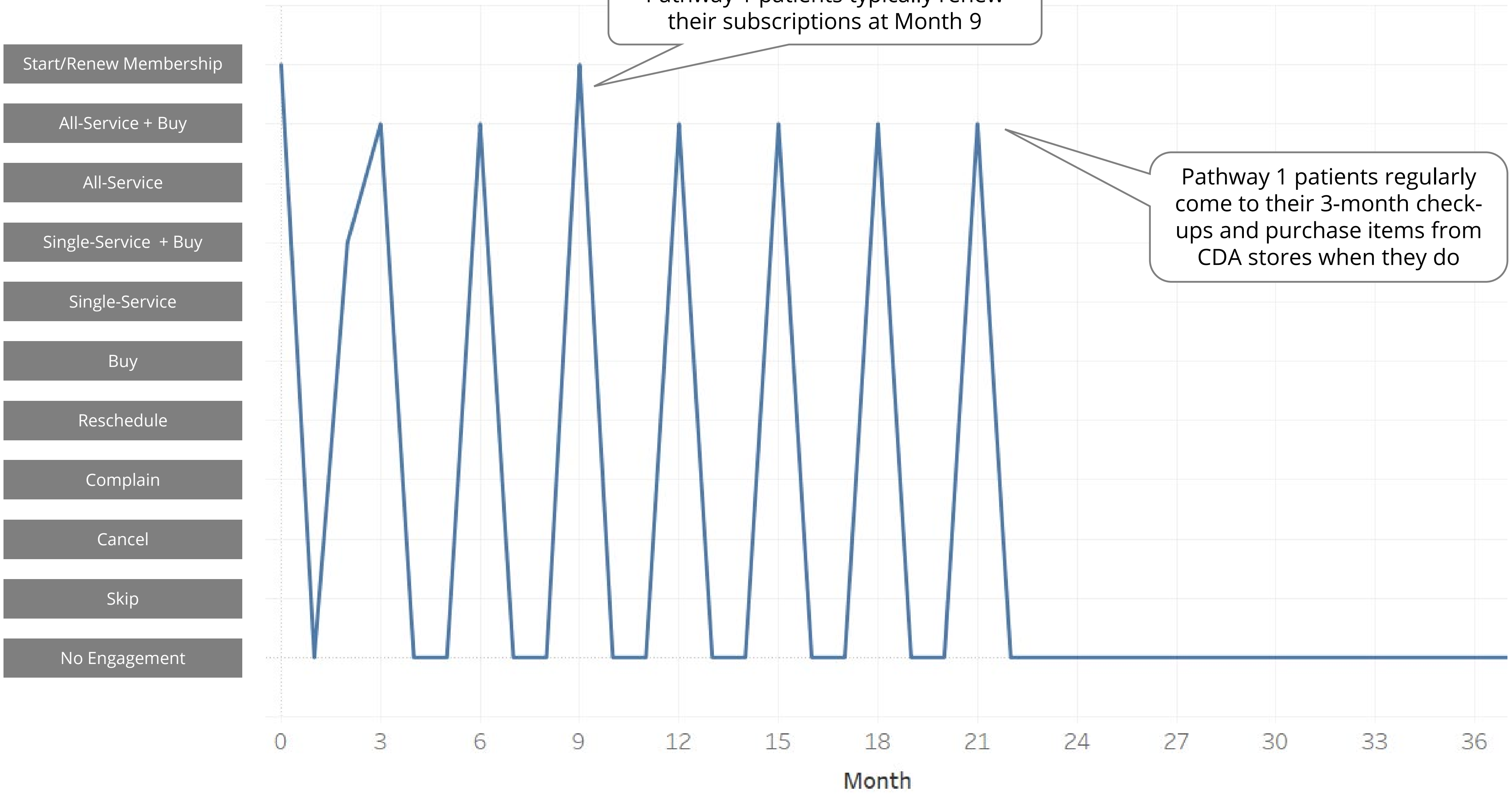


Tracking Patient Engagement Journeys

- Measure care events for each patient per month
- Index times to start at first appointment at CdA
- Create vectors where each element reflects highest level of engagement for one month
- Patients can be clustered by similar engagement patterns using k-medoids, and dissimilarity is pairwise distance between patient engagement vectors

Slide credit: Deloitte Consulting LLP

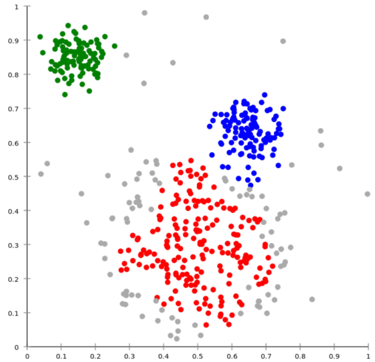
Typical Journey of High-Engagement Patient



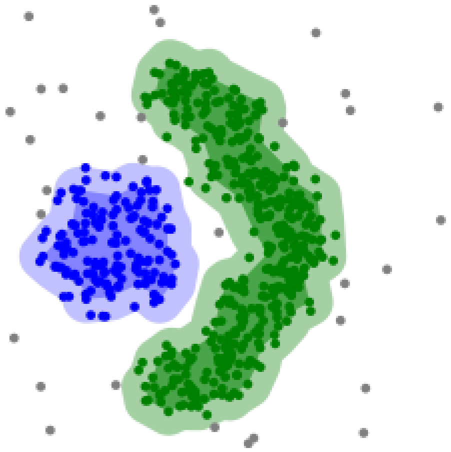
Pathway 1 patients typically renew their subscriptions at Month 9

Pathway 1 patients regularly come to their 3-month check-ups and purchase items from CDA stores when they do

Clustering Methods



- **K-means**: Numeric data, known k
- **Hierarchical clustering**: Unknown k
- **K-modes**: Categorical data
- **K-prototypes**: Categorical and numeric data
- **K-medoids**: Points in data set are seeds
- **DBScan**: Complex cluster shapes, noisy data



k-Means Clustering

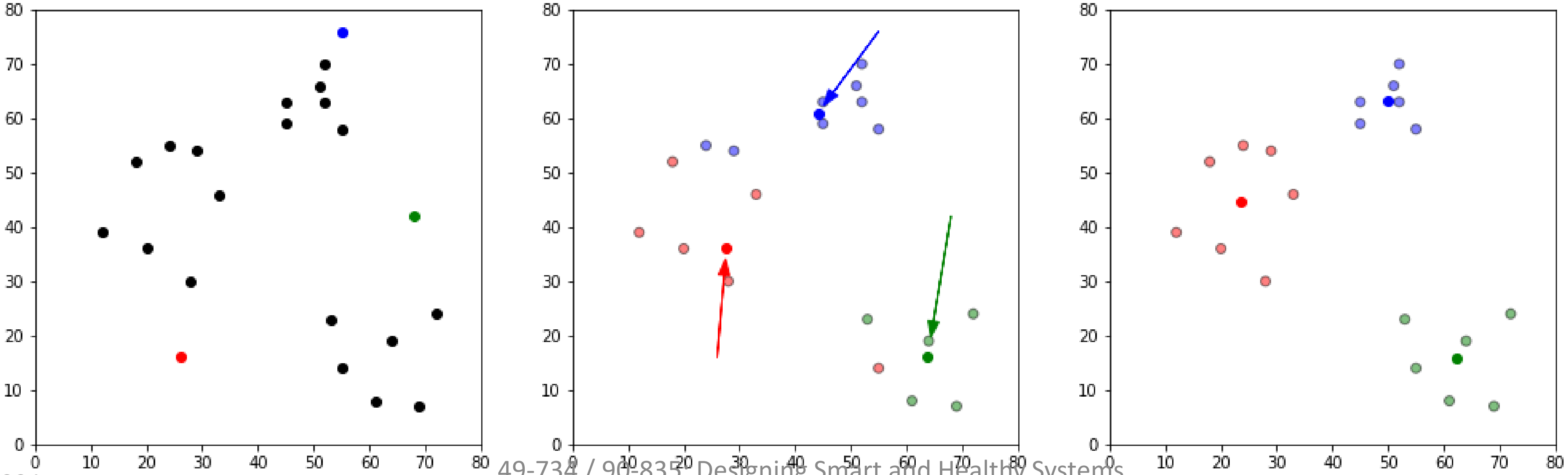
Partition *n* observations into *k* clusters.
3 steps:

- **Initialization** – *k* initial “means” (centroids) are generated at random
- **Assignment** – *k* clusters are created by associating each observation with the nearest centroid
- **Update** – The centroid of each clusters becomes its new mean

Assignment and Update are repeated iteratively until convergence

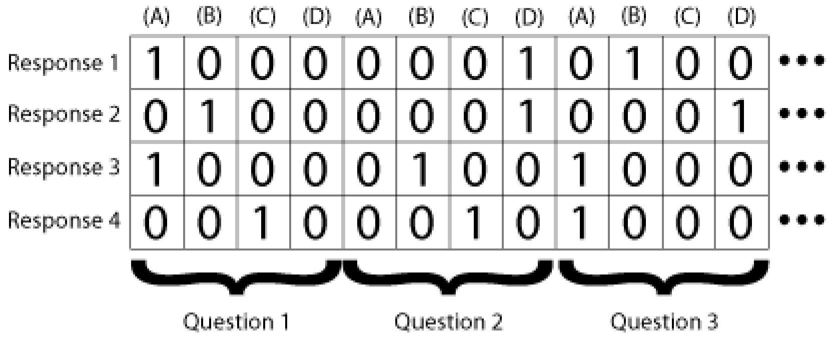
The end result is that the sum of squared errors is minimized between points and their respective centroids

http://benalexkeen.com/k-means-clustering-in-python/



K-modes

- K-means requires numeric data to calculate distances
- **K-modes** is useful for categorical data
  - Replaces distances with dissimilarities (number of total mismatches between two objects
  - Replaces the means in k-means with a vector where each element contains the category with the maximum number of matches across all the data. That is the centroid
  - Dissimilarity is the number of mismatches between that centroid vector and the vector representing a point observation



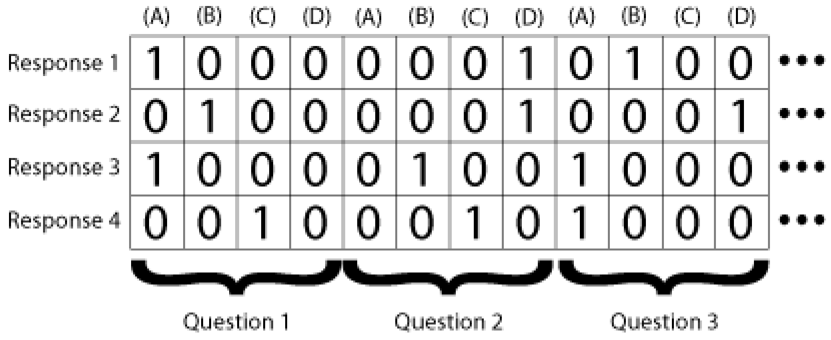
	(A)	(B)	(C)	(D)	(A)	(B)	(C)	(D)	(A)	(B)	(C)	(D)	
Response 1	1	0	0	0	0	0	0	1	0	1	0	0	•••
Response 2	0	1	0	0	0	0	0	1	0	0	0	1	•••
Response 3	1	0	0	0	0	1	0	0	1	0	0	0	•••
Response 4	0	0	1	0	0	0	1	0	1	0	0	0	•••
	Question 1				Question 2				Question 3				

Centroid is [A,D,A] if k = 1

Johnson, J. “K-modes”, posted in the Shape of Data blog, March 4, 2014, https://shapeofdata.wordpress.com/2014/03/04/k-modes/

K-modes

- K-means requires numeric data to calculate distances
- **K-modes** is useful for categorical data
  - Replaces distances with dissimilarities (number of total mismatches between two objects
  - Replaces the means in k-means with a vector where each element contains the category with the maximum number of matches across all the data. That is the centroid
  - Dissimilarity is the number of mismatches between that centroid vector and the vector representing a point observation



	Question 1				Question 2				Question 3				
	(A)	(B)	(C)	(D)	(A)	(B)	(C)	(D)	(A)	(B)	(C)	(D)	
Response 1	1	0	0	0	0	0	0	1	0	1	0	0	•••
Response 2	0	1	0	0	0	0	0	1	0	0	0	1	•••
Response 3	1	0	0	0	0	1	0	0	1	0	0	0	•••
Response 4	0	0	1	0	0	0	1	0	1	0	0	0	•••

Centroid is [A,D,A] if k = 1

Johnson, J. "K-modes", posted in the Shape of Data blog, March 4, 2014, https://shapeofdata.wordpress.com/2014/03/04/k-modes/

K-modes in Python

- May have to install k-modes using pip

```
import numpy as np
from kmodes.kmodes import KModes

# random categorical data
data = np.random.choice(20, (100, 10))

km = KModes(n_clusters=4, init='Huang',
n_init=5, verbose=1)

clusters = km.fit_predict(data)

# Print the cluster centroids
print(km.cluster_centroids_)
```

https://github.com/nicodv/kmodes

K-prototypes

- K-means and k-modes work for numeric and categorical data, respectively
- **K-prototypes** works for mixed data
  - Combines Euclidean distance for numeric attributes and dissimilarity from mode vector for categorical attributes
- In Python

```
from kmodes.kprototypes import KPrototypes

kproto = KPrototypes(n_clusters=15, init='Cao', verbose=2)
clusters = kproto.fit_predict(X, categorical=[1, 2])
```

Chamani Shiranthika, February 3, 2018, https://medium.com/datadriveninvestor/k-prototype-in-clustering-mixed-attributes-e6907db91914

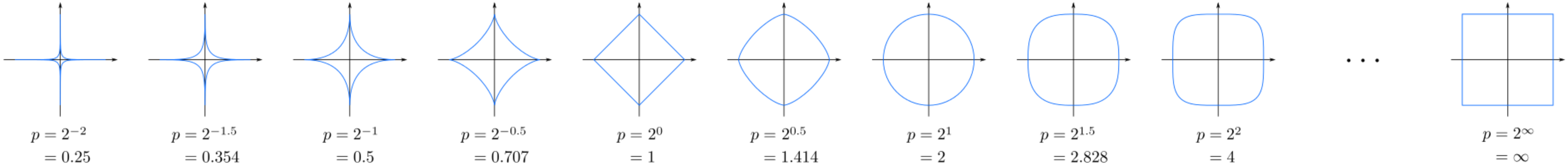
K-medoids

- K-means uses averages of coordinates as centroids; may not be points in the data set
- **K-medoids** uses points in the data set as "centroids," minimizing the sum of dissimilarities between objects labeled to be in a cluster and one of the objects (the *medoids*) designated as the representative of that cluster.
- Steps:
    1. **Initialization**: randomly select k of the m data points as the medoids
    2. **Assignment**: associate each data point with the closest medoid using **Minkowski distance** (generalization of Euclidean and Mahattan distance)
    3. **Update**: for each medoid j and each data point i associated with j, swap j and i and compute the total cost of the configuration (which is, the average dissimilarity of i to all the data points associated to j). Select the medoid j with the lowest cost of the configuration. Iterate between steps 2 and 3 until there is **no change** in the assignments.

Nguyen, T, October 24, 2019, https://towardsdatascience.com/k-medoids-clustering-on-iris-data-set-1931bf781e05

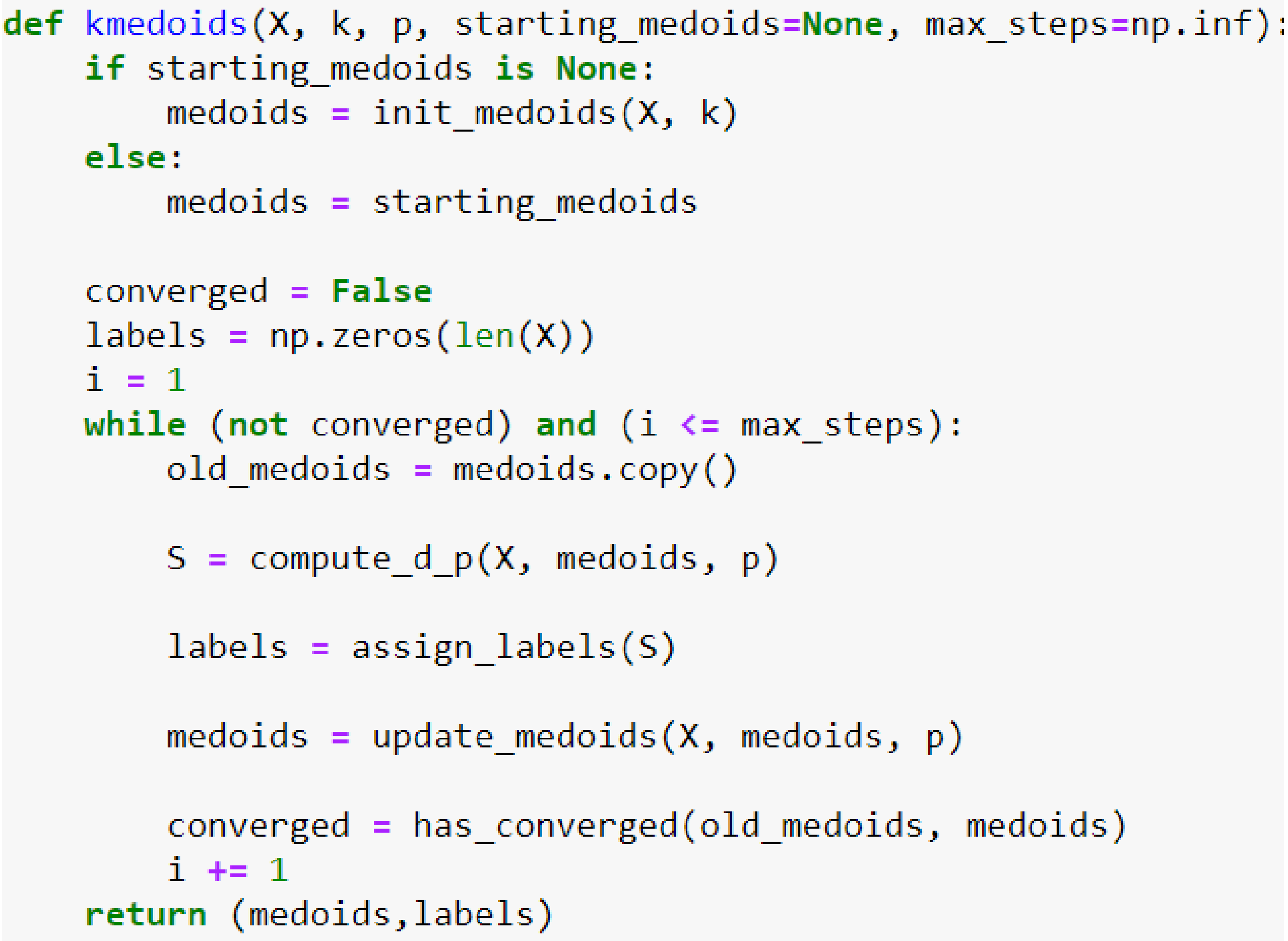
Minkowski Distance

- **Minkowski distance** is normed vector space of order p
  - Manhattan (grid) distance is Minkowski distance of order 1
  - Euclidean (air straight-line) distance is Minkowski distance of order 2 (also L2 distance)



https://en.wikipedia.org/wiki/Minkowski_distance

K-medoids in Python



```
def kmedoids(X, k, p, starting_medoids=None, max_steps=np.inf):
    if starting_medoids is None:
        medoids = init_medoids(X, k)
    else:
        medoids = starting_medoids

    converged = False
    labels = np.zeros(len(X))
    i = 1
    while (not converged) and (i <= max_steps):
        old_medoids = medoids.copy()

        S = compute_d_p(X, medoids, p)

        labels = assign_labels(S)

        medoids = update_medoids(X, medoids, p)

        converged = has_converged(old_medoids, medoids)
        i += 1
    return (medoids,labels)
```

Nguyen, T, October 24, 2019, https://towardsdatascience.com/k-medoids-clustering-on-iris-data-set-1931bf781e05

Clustering patient journeys using K-medoids

1. Create a matrix in which each row is a patient and each column represents some time period in a sequence, where the first value in the sequence for a patient starts with their first appointment at CdA.
2. Compute the value in each cell of the matrix i,j as the value of some outcome variable for patient i at time period j. So if the number of appointments scheduled is the outcome variable, the row for Patient A who had his or her first appointment in Jan 2014 might have the number of appointments for Jan 2014, Feb 2014, etc.
3. Cluster patients by similar sequences of engagement using k-medoids. Similarity is measured by pairwise distance between patient engagement vectors

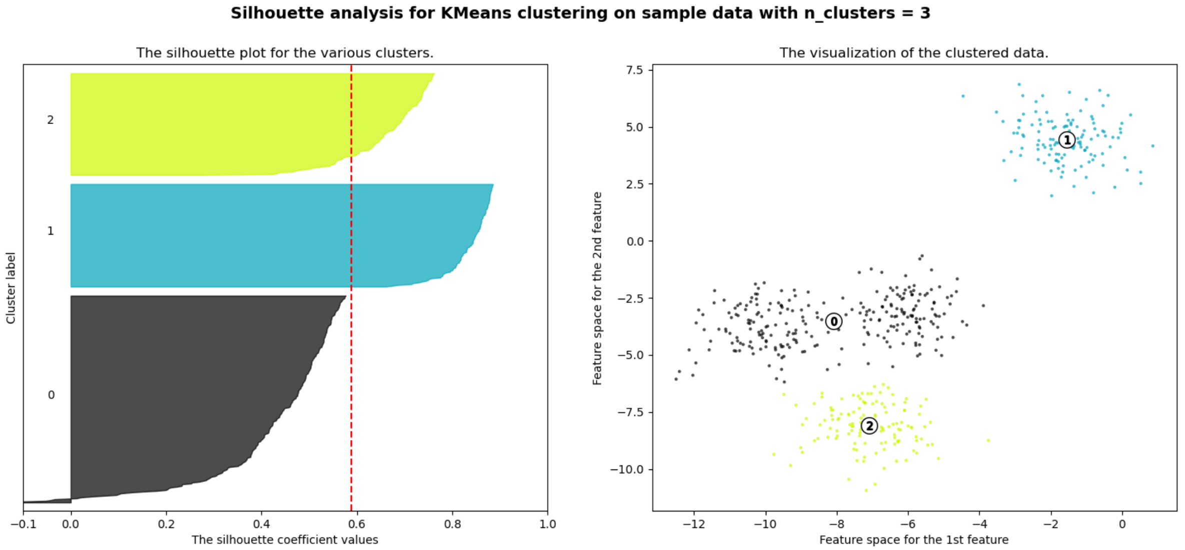
Evaluating Clustering Outcomes

- Internal evaluations
  - Silhouette method, average inter-cluster separation
  - Dunn index: min(inter-cluster distances) / max (intra-cluster distances)
- External evaluations, when known class labels are available, e.g. F-measure (balances precision, recall)
- Cluster tendency, to evaluate if data should cluster at all, e.g. Hopkins statistic (.5 if uniform, 1 if clustered)

$$H = \frac{\sum_{i=1}^{m} u_i^d}{\sum_{i=1}^{m} u_i^d + \sum_{i=1}^{m} w_i^d} \, ,$$

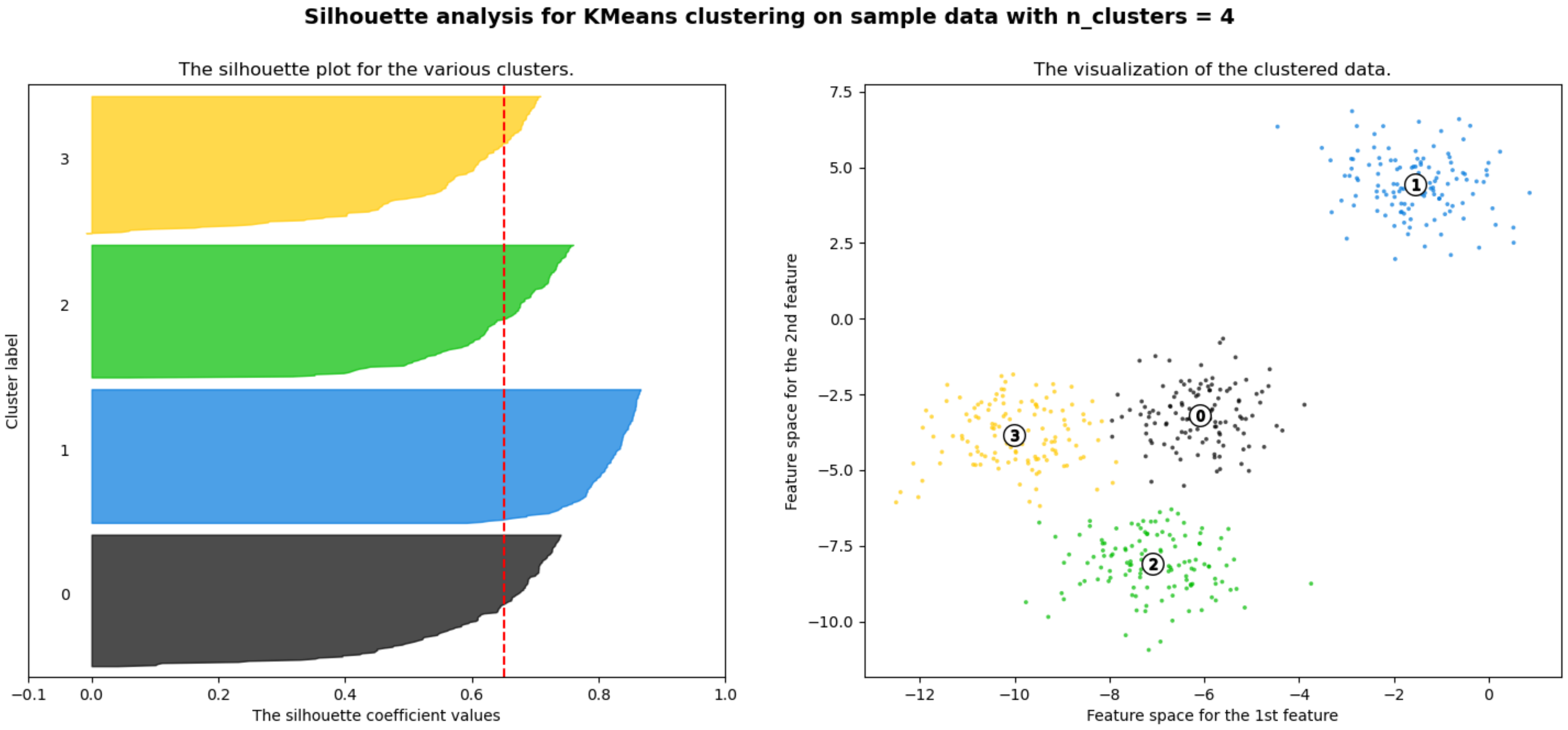
Silhouette Method for Choosing k (k=3)

- Silhouette coefficients measure separation of points from other clusters +1 far away, 0 on the decision boundary, -1 might be in wrong cluster



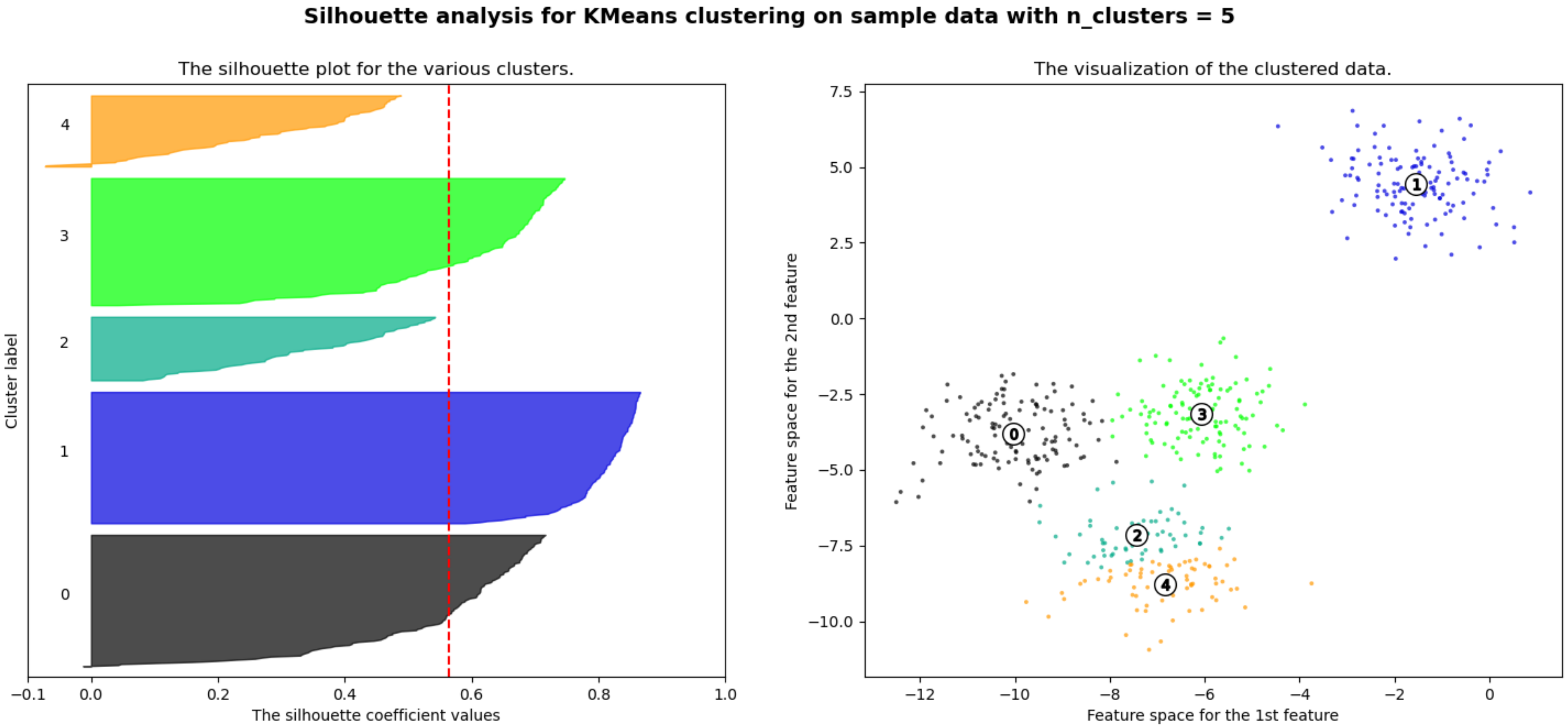
Sci-kit learn documentation, “Selecting the number of clusters with silhouette analysis on KMeans clustering”, https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Silhouette Method for Choosing k (k=4)



Sci-kit learn documentation, “Selecting the number of clusters with silhouette analysis on KMeans clustering”, https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Silhouette Method for Choosing k (k=5)

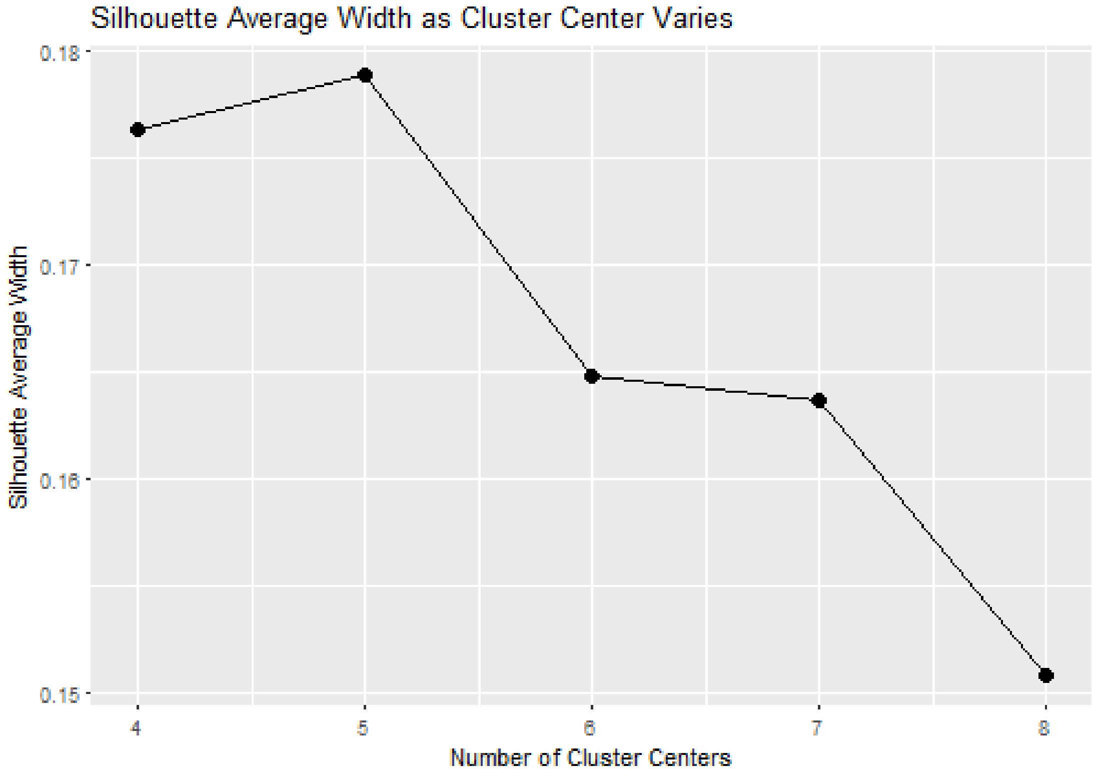


Sci-kit learn documentation, "Selecting the number of clusters with silhouette analysis on KMeans clustering", https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Silhouette Method to Determine Optimal k

Silhouette measures how similar each point is to other points in the same cluster and dissimilarity to points in other clusters

https://en.wikipedia.org/wiki/Silhouette_(clustering)



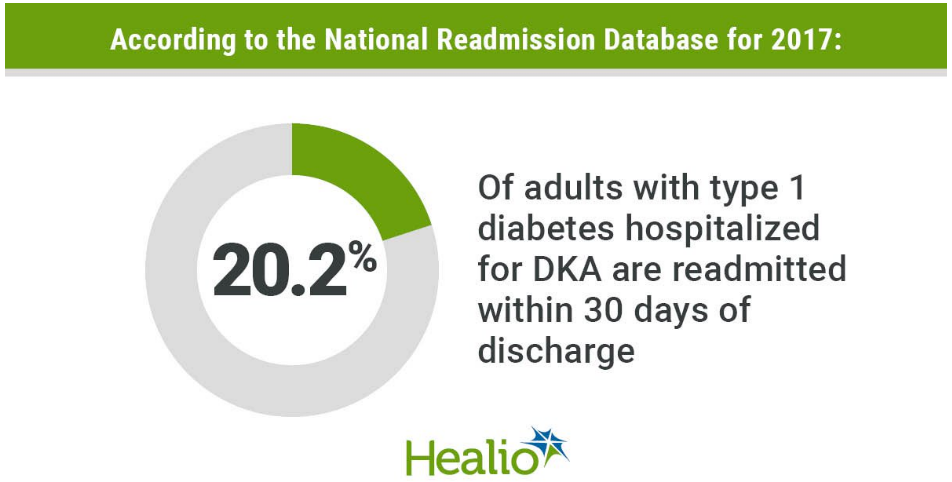
http://www.business-science.io/business/2016/08/07/CustomerSegmentationPt1.html

Prediction methods

- Linear models
  - Least Absolute Shrinkage and Selection Operator (LASSO Regression)
  - Ridge regressions
- Support vector regression
- K-nearest neighbors regression
- Tree-based methods
  - Decision tree
  - Random forest
  - Boosting
    - eXtreme Gradient Boosting (XGBoost)

Reducing Unplanned Hospital Readmissions

- One fifth of all patients with diabetic keto-acidosis (DKA) are readmitted within 30 days (2017)
- American hospitals spent over $41B on diabetic patients readmitted within 30 days of discharge (2011)
- Many readmissions are preventable: Estimates range from 20 percent up the 75 percent (Department of Health and Human Services)



According to the National Readmission Database for 2017:

20.2%

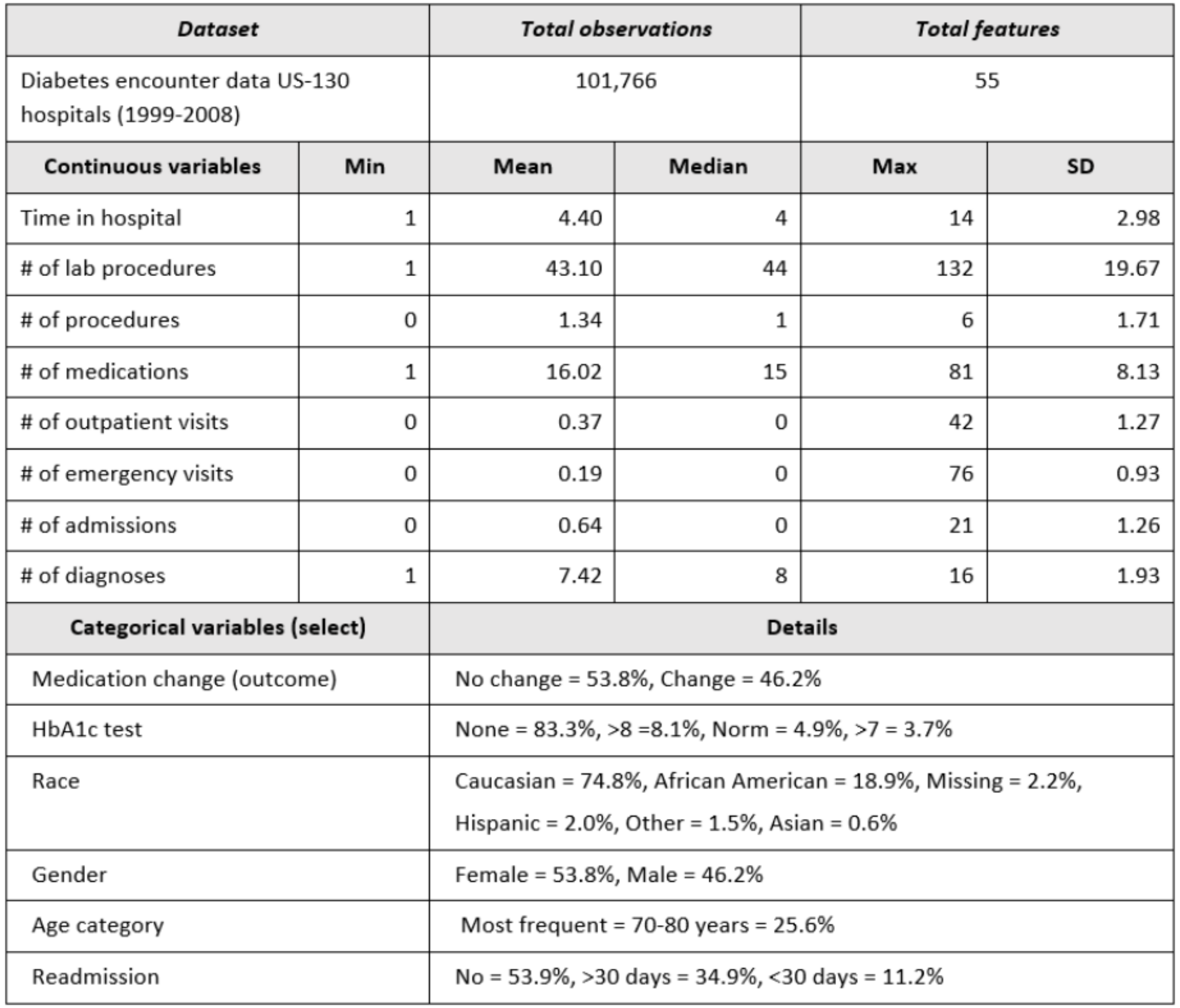
Of adults with type 1 diabetes hospitalized for DKA are readmitted within 30 days of discharge

Healio

https://www.healio.com/news/endocrinology/20210325/hospital-readmission-for-dka-associated-with-increased-mortality-risk-in-type-1-diabetes

Readmissions Data

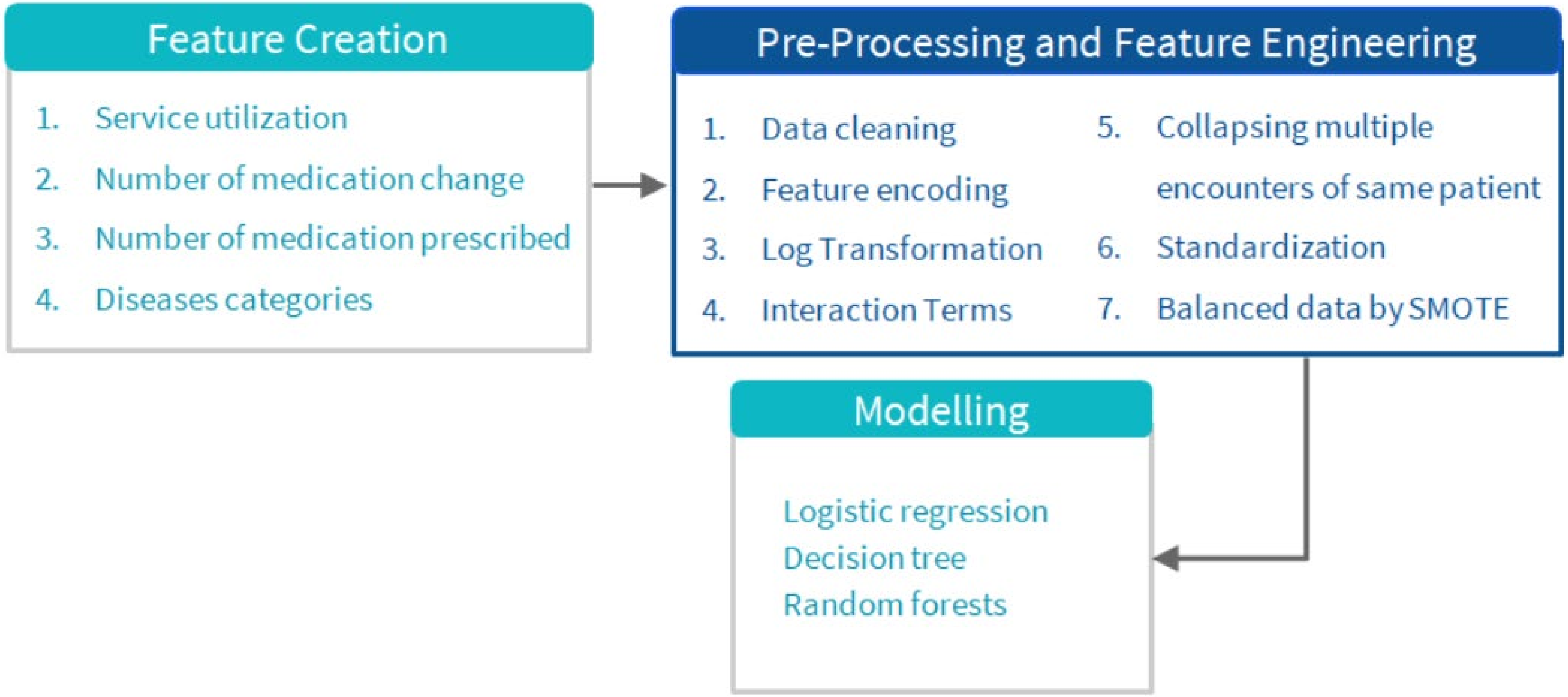
- Data from 130 hospitals on diabetes readmissions publicly available from UCI ML repository https://archive.ics.uci.edu/ml/datasets/diabetes+130-us+hospitals+for+years+1999-2008#
- 101K observations 55 features including patient characteristics, conditions, tests, and 23 medications
- Use data to answer
  - What factors are the strongest predictors of hospital readmission in diabetic patients?
  - How well can we predict hospital readmission in this dataset with limited features?



Dataset		***Total observations***		***Total features***	
Diabetes encounter data US-130 hospitals (1999-2008)		101,766		55	
Continuous variables	**Min**	**Mean**	**Median**	**Max**	**SD**
Time in hospital	1	4.40	4	14	2.98
# of lab procedures	1	43.10	44	132	19.67
# of procedures	0	1.34	1	6	1.71
# of medications	1	16.02	15	81	8.13
# of outpatient visits	0	0.37	0	42	1.27
# of emergency visits	0	0.19	0	76	0.93
# of admissions	0	0.64	0	21	1.26
# of diagnoses	1	7.42	8	16	1.93
Categorical variables (select)		**Details**			
Medication change (outcome)		No change = 53.8%, Change = 46.2%			
HbA1c test		None = 83.3%, >8 =8.1%, Norm = 4.9%, >7 = 3.7%			
Race		Caucasian = 74.8%, African American = 18.9%, Missing = 2.2%, Hispanic = 2.0%, Other = 1.5%, Asian = 0.6%			
Gender		Female = 53.8%, Male = 46.2%			
Age category		Most frequent = 70-80 years = 25.6%			
Readmission		No = 53.9%, >30 days = 34.9%, <30 days = 11.2%			

Raza, U. “How to use machine learning to predict hospital readmissions? (Part 1)”, January 9, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-1-bd137cbdba07

Process of Applying ML to Readmissions Data



Feature Creation

1. Service utilization
2. Number of medication change
3. Number of medication prescribed
4. Diseases categories

Pre-Processing and Feature Engineering

1. Data cleaning
2. Feature encoding
3. Log Transformation
4. Interaction Terms
5. Collapsing multiple encounters of same patient
6. Standardization
7. Balanced data by SMOTE

Modelling

Logistic regression
Decision tree
Random forests

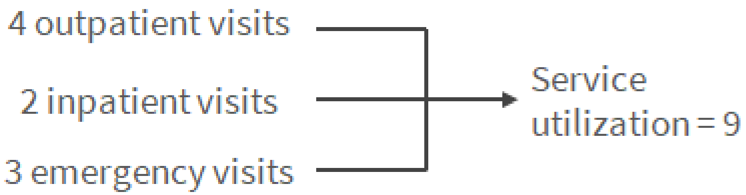
Raza, U. “How to use machine learning to predict hospital readmissions? (Part 1)”, January 9, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-1-bd137cbdba07

Results of Exploratory Data Analysis

- Missing values
  - race 2273
  - weight 98569
    - Drop the field because 98% of values missing
  - payer_code 40256, medical_specialty 49949
    - Drop the fields because 40%-50% missing values
  - diag_1 21, diag_2 358, diag_3 1423
    - Drop the record only where all three diagnoses missing
  - gender 3
    - Drop the records
- Drop records where patient has died
- Drop fields citoglipton and examide because all have same value

Raza, U. “How to use machine learning to predict hospital readmissions? (Part 1)”, January 9, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-1-bd137cbdba07

Creating New Variables



- Service utilization
- Medication changes (how many made in total)
- Collapse three diagnoses codes with potential values of 700-900 ICD codes each into 9 disease categories: *Circulatory, Respiratory, Digestive, Diabetes, Injury, Musculoskeletal, Genitourinary, Neoplasms, and Others*
- Collapse categories for admission source, admission type and discharge disruption.
- Other recodings (strings to binary values, outcome to binary, age categories to bin midpoints, collapse multiple encounters for same patient)

Raza, U. "How to use machine learning to predict hospital readmissions? (Part 1)", January 9, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-1-bd137cbdba07

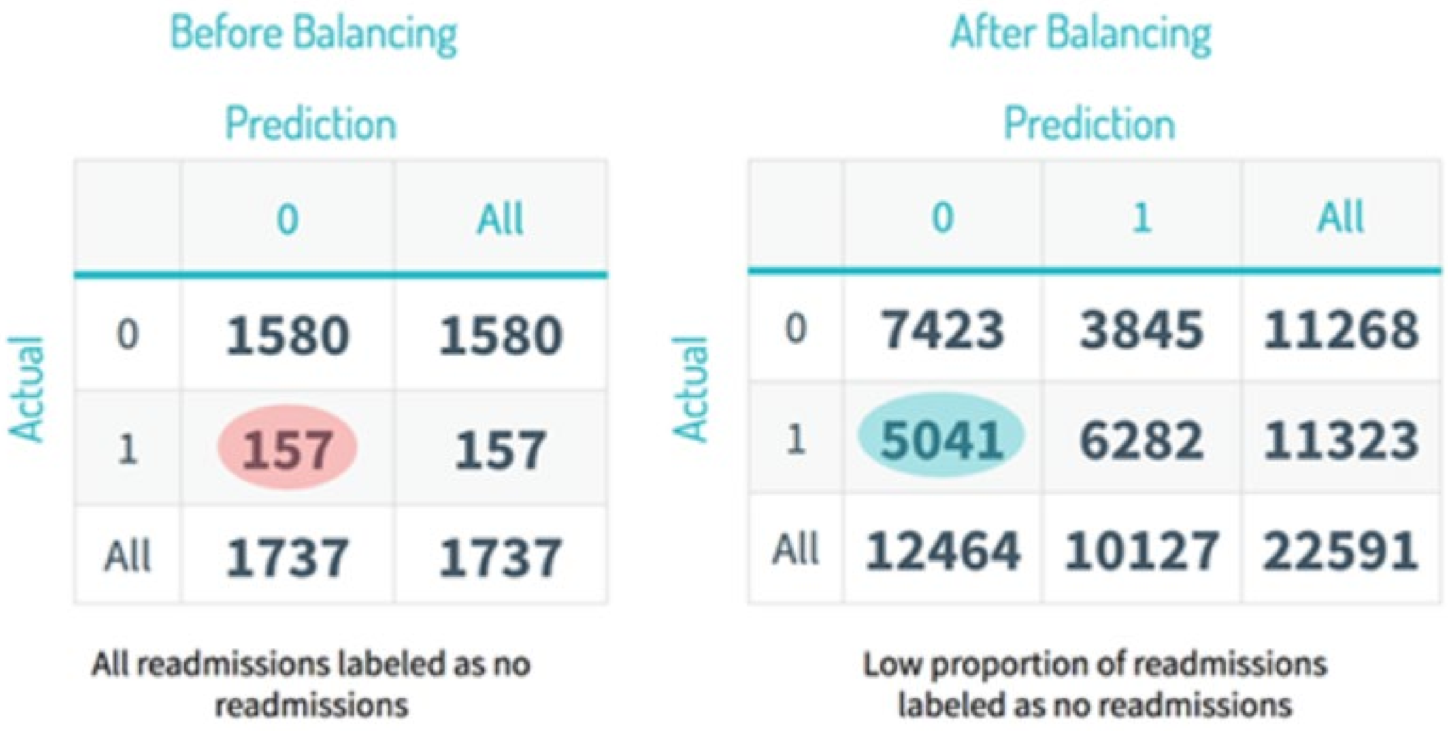
Other Transformations

- Logs to remove skew: number of emergency visits, service utilization, number of inpatient admissions and number of outpatient visits
- Standardization for all numeric features: Z-score
- Outlier removal: Removed all data outside 3 standard deviations
- Interaction terms for correlated variables that have interdependent effects.

Raza, U. "How to use machine learning to predict hospital readmissions? (Part 1)", January 9, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-1-bd137cbdba07

The Importance of Balanced Data

- If 90% of patients are not readmitted, a model that predicts no patients will be readmitted will be 90% accurate!
- SMOTE (Synthetic Minority Oversampling Technique) oversamples underrepresented class of readmissions to balance the data set



Before Balancing

Prediction

		0	All
Actual	0	1580	1580
	1	157	157
	All	1737	1737

All readmissions labeled as no readmissions

After Balancing

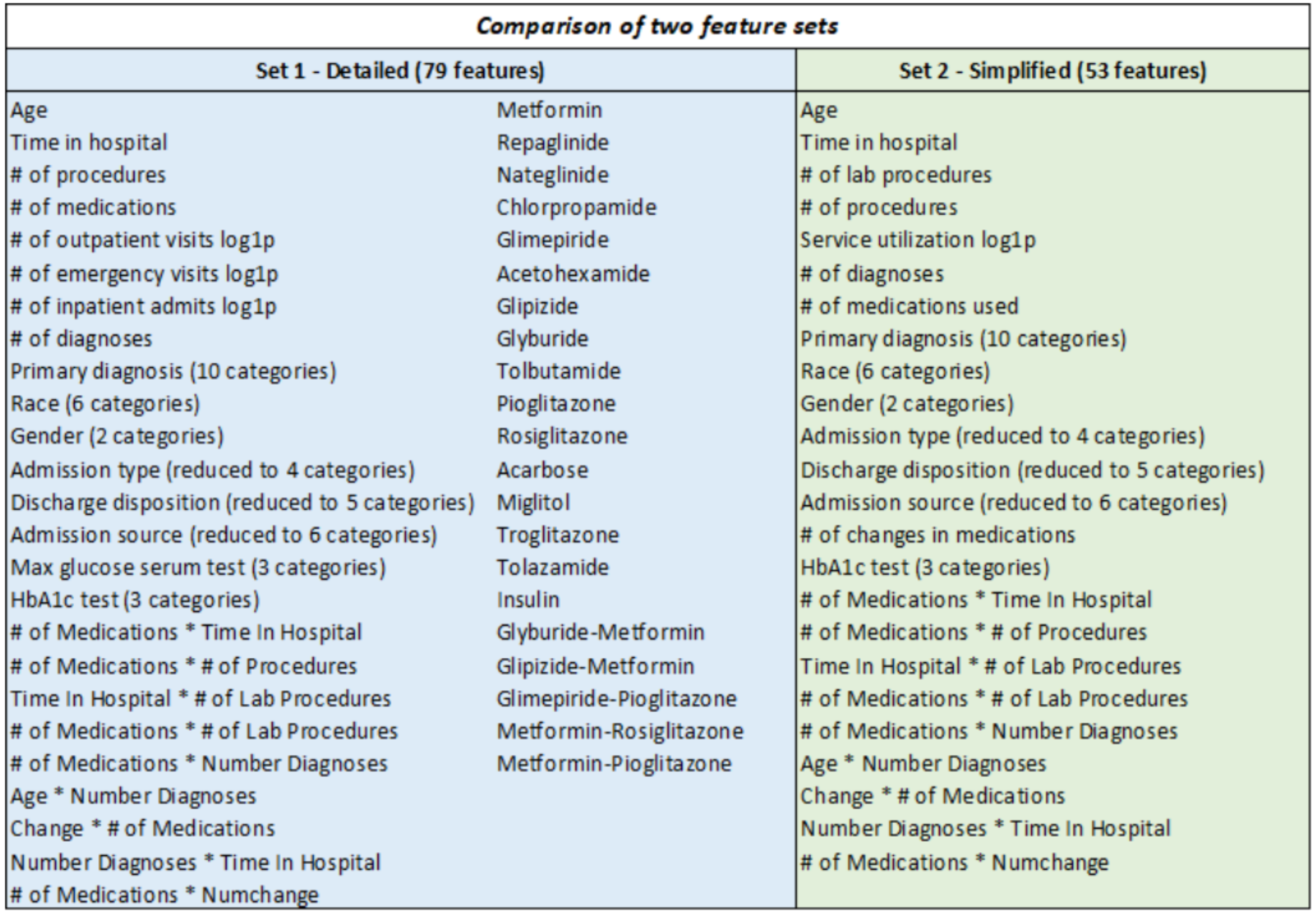
Prediction

		0	1	All
Actual	0	7423	3845	11268
	1	5041	6282	11323
	All	12464	10127	22591

Low proportion of readmissions labeled as no readmissions

Raza, U. “How to use machine learning to predict hospital readmissions? (Part 1)”, January 9, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-1-bd137cbdba07

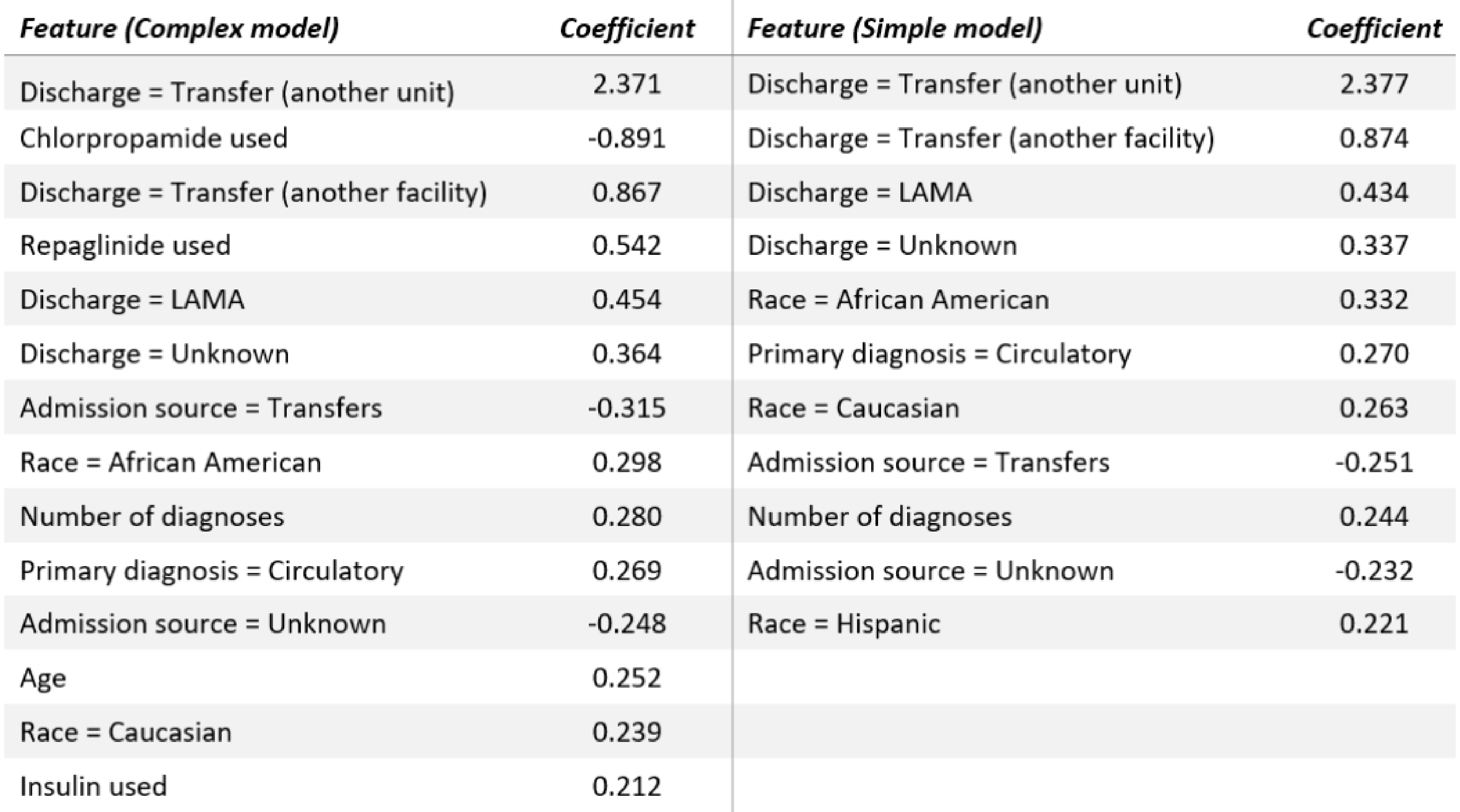
Feature Selection



Comparison of two feature sets		
Set 1 - Detailed (79 features)		**Set 2 - Simplified (53 features)**
Age	Metformin	Age
Time in hospital	Repaglinide	Time in hospital
# of procedures	Nateglinide	# of lab procedures
# of medications	Chlorpropamide	# of procedures
# of outpatient visits log1p	Glimepiride	Service utilization log1p
# of emergency visits log1p	Acetohexamide	# of diagnoses
# of inpatient admits log1p	Glipizide	# of medications used
# of diagnoses	Glyburide	Primary diagnosis (10 categories)
Primary diagnosis (10 categories)	Tolbutamide	Race (6 categories)
Race (6 categories)	Pioglitazone	Gender (2 categories)
Gender (2 categories)	Rosiglitazone	Admission type (reduced to 4 categories)
Admission type (reduced to 4 categories)	Acarbose	Discharge disposition (reduced to 5 categories)
Discharge disposition (reduced to 5 categories)	Miglitol	Admission source (reduced to 6 categories)
Admission source (reduced to 6 categories)	Troglitazone	# of changes in medications
Max glucose serum test (3 categories)	Tolazamide	HbA1c test (3 categories)
HbA1c test (3 categories)	Insulin	# of Medications * Time In Hospital
# of Medications * Time In Hospital	Glyburide-Metformin	# of Medications * # of Procedures
# of Medications * # of Procedures	Glipizide-Metformin	Time In Hospital * # of Lab Procedures
Time In Hospital * # of Lab Procedures	Glimepiride-Pioglitazone	# of Medications * # of Lab Procedures
# of Medications * # of Lab Procedures	Metformin-Rosiglitazone	# of Medications * Number Diagnoses
# of Medications * Number Diagnoses	Metformin-Pioglitazone	Age * Number Diagnoses
Age * Number Diagnoses		Change * # of Medications
Change * # of Medications		Number Diagnoses * Time In Hospital
Number Diagnoses * Time In Hospital		# of Medications * Numchange
# of Medications * Numchange		

Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

Feature Weights in Logistic Regression



Feature (Complex model)	*Coefficient*	*Feature (Simple model)*	*Coefficient*
Discharge = Transfer (another unit)	2.371	Discharge = Transfer (another unit)	2.377
Chlorpropamide used	-0.891	Discharge = Transfer (another facility)	0.874
Discharge = Transfer (another facility)	0.867	Discharge = LAMA	0.434
Repaglinide used	0.542	Discharge = Unknown	0.337
Discharge = LAMA	0.454	Race = African American	0.332
Discharge = Unknown	0.364	Primary diagnosis = Circulatory	0.270
Admission source = Transfers	-0.315	Race = Caucasian	0.263
Race = African American	0.298	Admission source = Transfers	-0.251
Number of diagnoses	0.280	Number of diagnoses	0.244
Primary diagnosis = Circulatory	0.269	Admission source = Unknown	-0.232
Admission source = Unknown	-0.248	Race = Hispanic	0.221
Age	0.252		
Race = Caucasian	0.239		
Insulin used	0.212		

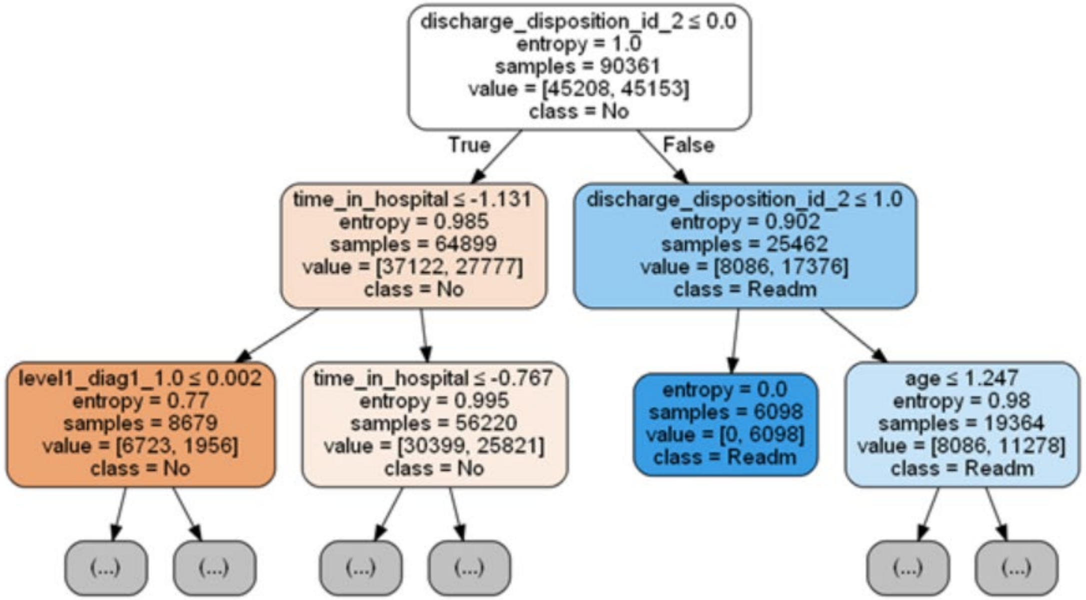
Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

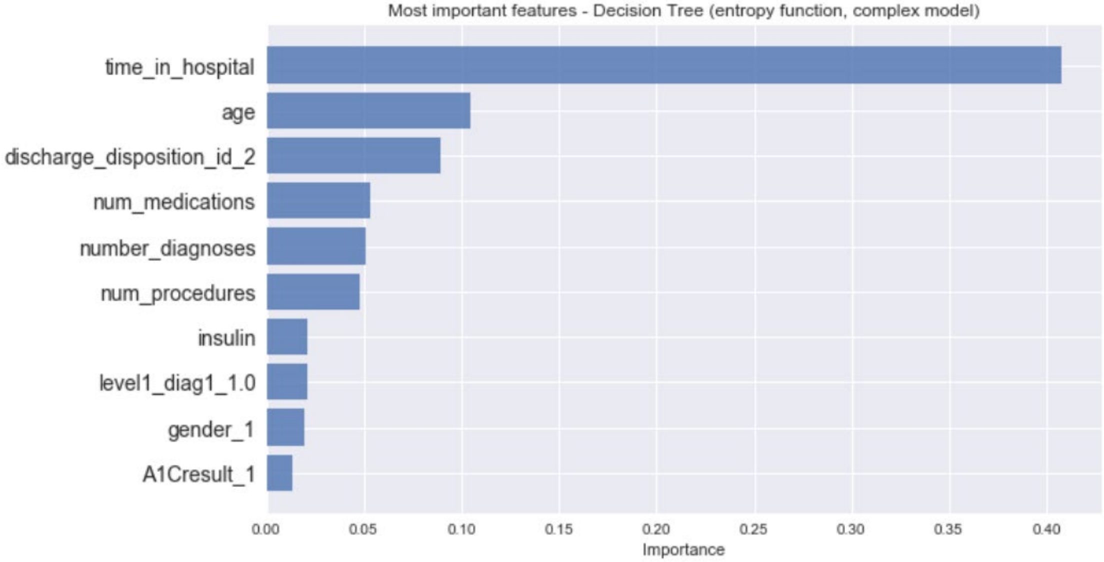
Logistic Regression Results

```
Y_dev_predict = logreg.predict(X_dev)
pd.crosstab(pd.Series(Y_dev, name = 'Actual'), pd.Series(Y_dev_predict,
name = 'Predict'), margins = True)
from sklearn.metrics import accuracy_score, precision_score,
recall_score, roc_auc_score
print("Accuracy is {0:.2f}".format(accuracy_score(Y_dev,
Y_dev_predict)))
print("Precision is {0:.2f}".format(precision_score(Y_dev,
Y_dev_predict)))
print("Recall is {0:.2f}".format(recall_score(Y_dev, Y_dev_predict)))
print("AUC is {0:.2f}".format(roc_auc_score(Y_dev,
Y_dev_predict)))Accuracy is 0.61
Precision is 0.62
Recall is 0.55
AUC is 0.61
```

Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

Decision Tree Results and Feature Importance





Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

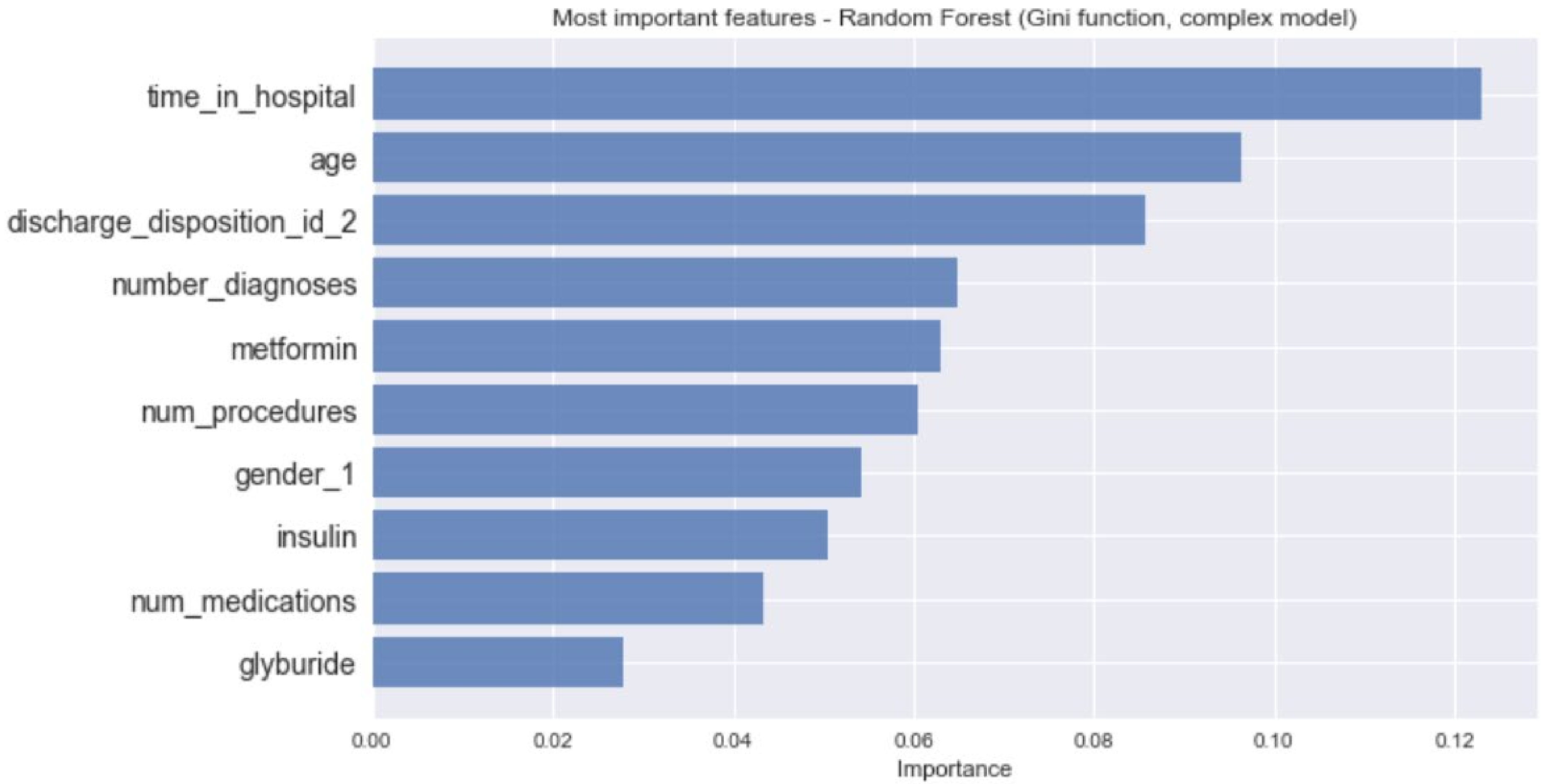
Decision Tree Results

```
Y_dev_predict = dte.predict(X_dev)
pd.crosstab(pd.Series(Y_dev, name = 'Actual'), pd.Series(Y_dev_predict,
name = 'Predict'), margins = True)
from sklearn.metrics import accuracy_score, precision_score,
recall_score, roc_auc_score
print("Accuracy is {0:.2f}".format(accuracy_score(Y_dev, Y_dev_predict)))
print("Precision is {0:.2f}".format(precision_score(Y_dev,
Y_dev_predict)))
print("Recall is {0:.2f}".format(recall_score(Y_dev, Y_dev_predict)))
print("AUC is {0:.2f}".format(roc_auc_score(Y_dev, Y_dev_predict)))
```

```
Accuracy is 0.91
Precision is 0.93
Recall is 0.89
AUC is 0.91
```

Raza, U. "How to use machine learning to predict hospital readmissions? (Part 2)", January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

Random Forest Feature Importance



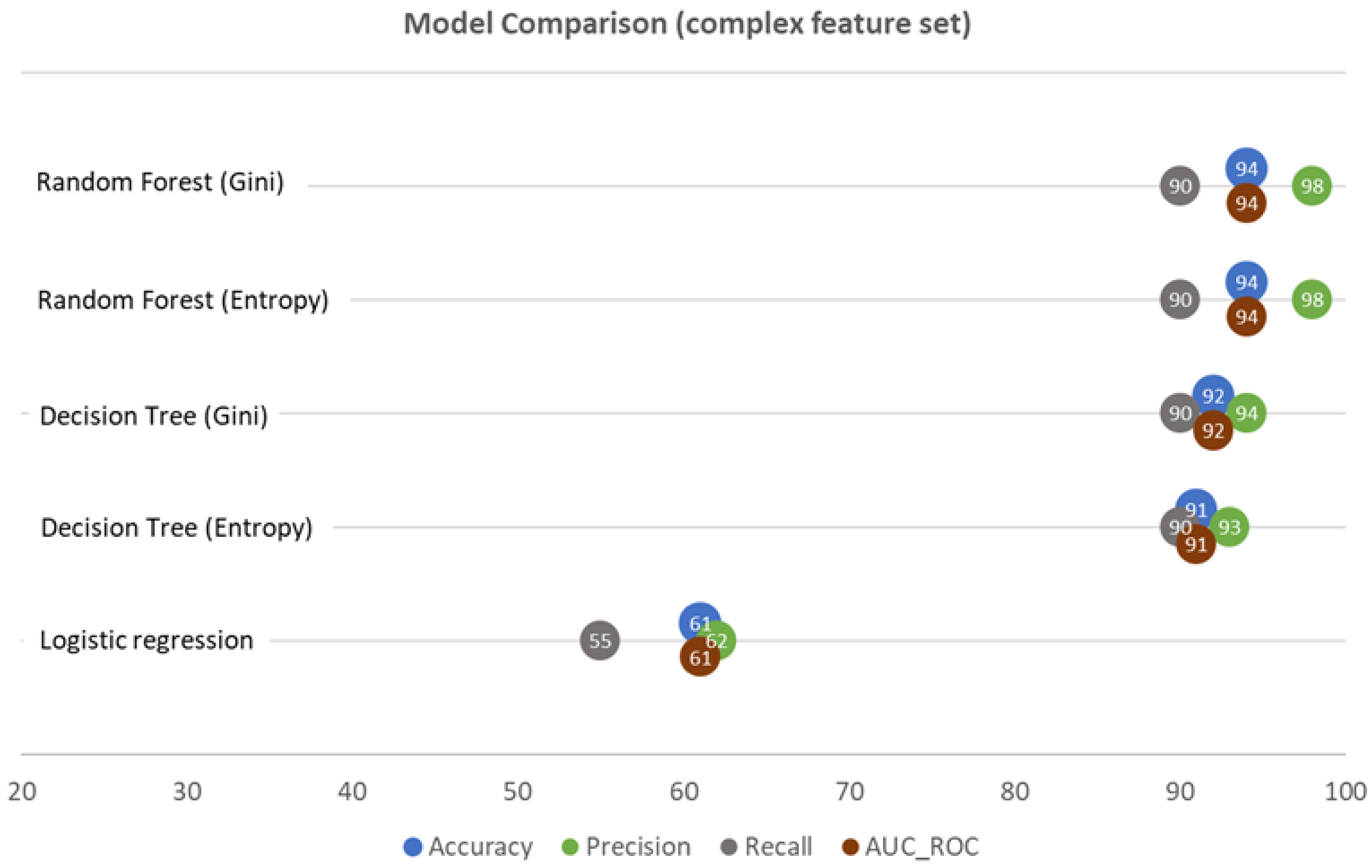
Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

Choice of Models in Readmissions Prediction

- Used due to combination of human interpretability and accuracy:
  - Logistic regression (including Lasso to select features)
  - Decision trees
  - Random Forests
- Rejected:
  - K-Nearest neighbors, due to equal weight
  - Support Vector Machines, due to limited interpretability
  - Neural networks, due to limited interpretability (and data requirements)

Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

Model Comparison

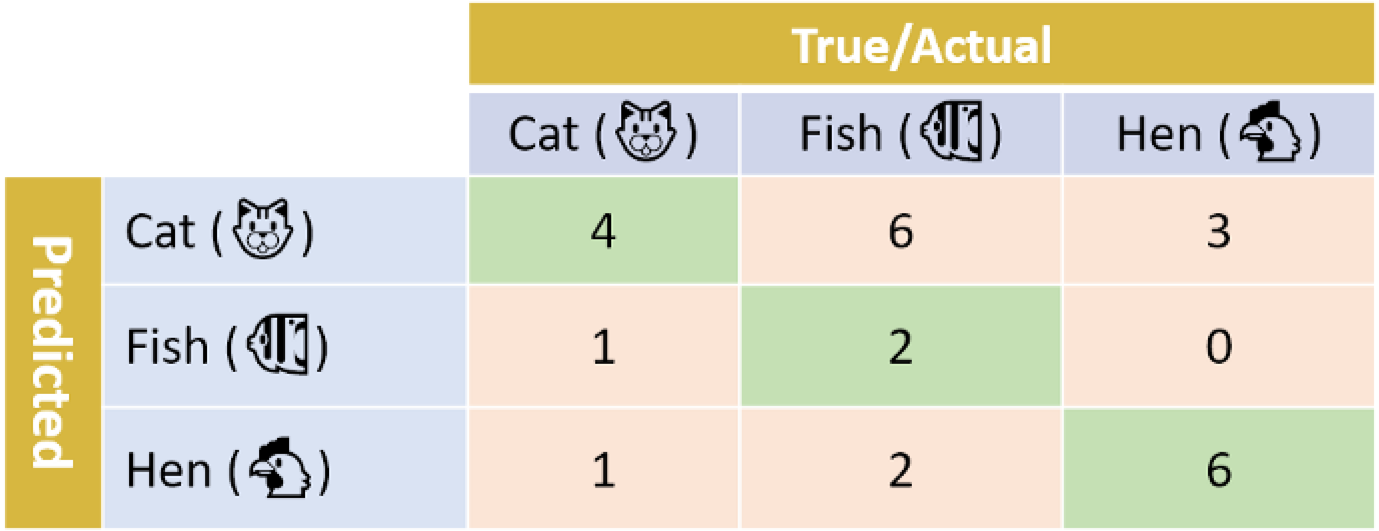


Raza, U. “How to use machine learning to predict hospital readmissions? (Part 2)”, January 16, 2018 https://medium.com/berkeleyischool/how-to-use-machine-learning-to-predict-hospital-readmissions-part-2-616a0c920ab1

Visualizing Prediction Results

- Evaluations
    - R-square
    - Area Under (ROC) Curve
    - F1-micro-average
- Visualizations
    - Scatterplot
    - ROC curve
    - Confusion matrix

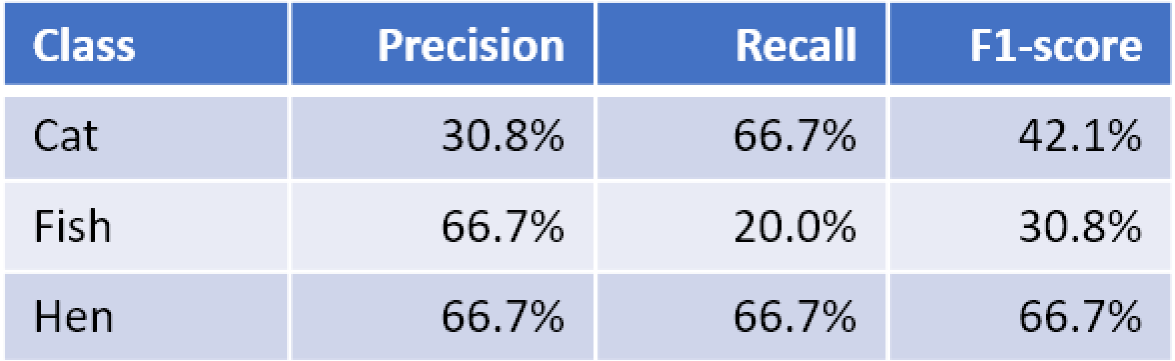
Multi-class Confusion Matrix



		True/Actual		
		Cat (🐱)	Fish (🐟)	Hen (🐔)
Predicted	Cat (🐱)	4	6	3
	Fish (🐟)	1	2	0
	Hen (🐔)	1	2	6

B. Shmueli, July 3, 2019 https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-ebe8b2c2ca1

Per Class Precision, Recall, and F1 Score: Macro-averages

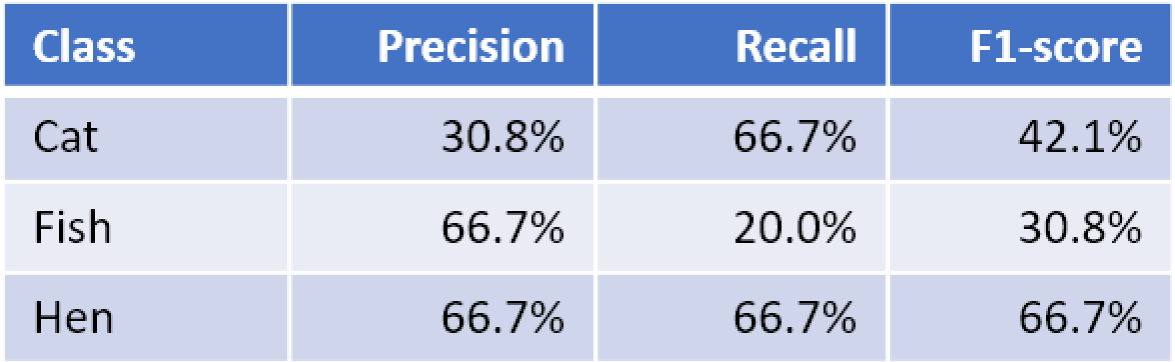


Class	Precision	Recall	F1-score
Cat	30.8%	66.7%	42.1%
Fish	66.7%	20.0%	30.8%
Hen	66.7%	66.7%	66.7%

- **Macro-precision** = (31% + 67% + 67%) / 3 = 54.7%
- **Macro-recall** = (67% + 20% + 67%) / 3 = 51.1%
- **Macro-F1** = (42.1% + 30.8% + 66.7%) / 3 = 46.5%

B. Shmueli, July 3, 2019 https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-ebe8b2c2ca1

Per Class Precision, Recall, and F1 Score: Micro-averages



Class	Precision	Recall	F1-score
Cat	30.8%	66.7%	42.1%
Fish	66.7%	20.0%	30.8%
Hen	66.7%	66.7%	66.7%

- **Weighted-precision**=(6 × 30.8% + 10 × 66.7% + 9 × 66.7%)/25 = 58.1%
- **Weighted-recall** = (6 × 66.7% + 10 × 20.0% + 9 × 66.7%) / 25 = 48.0%
- **Weighted-F1** = (6 × 42.1% + 10 × 30.8% + 9 × 66.7%) / 25 = 46.4%

B. Shmueli, July 3, 2019 https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-ebe8b2c2ca1

Concluding Remarks

- Clustering is not just k-means
- Sequences of observations can be clustered
- Prediction requires substantial data preparation
- **Next class (November 4):** Trajectory Analysis (Prof. Daniel Nagin)