



Introduction to **Audio Content Analysis**

module 4.2: regression & clustering

alexander lerch

introduction

overview

corresponding textbook section

sections 4.2 – 4.4

■ lecture content

- regression: non-categorical data analysis
- clustering: unsupervised data analysis

■ learning objectives

- describe the basic principles of data-driven machine learning approaches
- implement linear regression in Python
- implement kMeans clustering in Python



introduction

overview

corresponding textbook section

sections 4.2 – 4.4

■ lecture content

- regression: non-categorical data analysis
- clustering: unsupervised data analysis

■ learning objectives

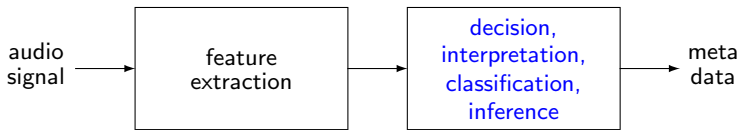
- describe the basic principles of data-driven machine learning approaches
- implement linear regression in Python
- implement kMeans clustering in Python



regression

introduction

remember the flow chart of a general ACA system:



■ *classification:*

- assign class labels to data

■ *regression:*

- estimate numerical labels for data

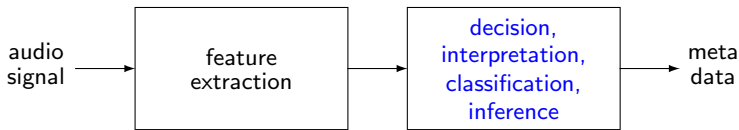
■ *clustering:*

- find grouping patterns in data

regression

introduction

remember the flow chart of a general ACA system:



■ *classification*:

- assign class labels to data

■ *regression*:

- estimate numerical labels for data

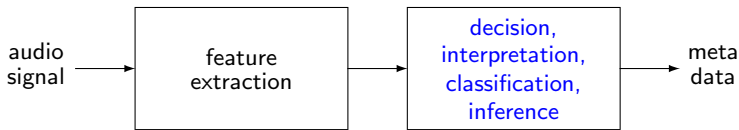
■ *clustering*:

- find grouping patterns in data

regression

introduction

remember the flow chart of a general ACA system:



■ *classification:*

- assign class labels to data

■ *regression:*

- estimate numerical labels for data

■ *clustering:*

- find grouping patterns in data

regression

introduction

- given a set of pairs of data and corresponding output observations
- find model that maps input to output
- model can then be used to predict (continuous value) output for an unknown new input

regression

introduction

- given a set of pairs of data and corresponding output observations
- find model that maps input to output
- model can then be used to predict (continuous value) output for an unknown new input

regression

linear regression

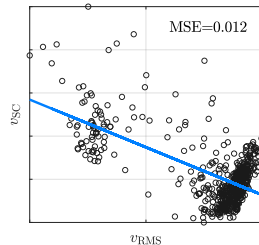
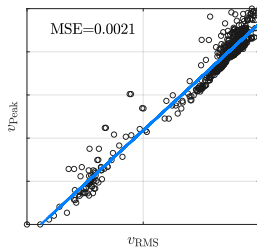
- estimate the slope m and offset b of a straight line that fits the data best:

$$\hat{y}(r) = m \cdot v(r) + b$$

- minimizing the mean squared error leads to:

$$b = \mu_y - m \cdot \mu_v$$

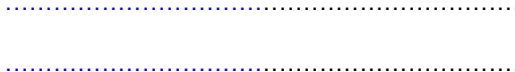
$$m = \frac{\sum_{r=0}^{\mathcal{R}-1} (y(r) - \mu_y) \cdot (v(r) - \mu_v)}{\sum_{r=0}^{\mathcal{R}-1} (v(r) - \mu_v)^2}$$



clustering

introduction

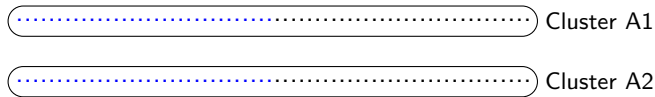
- clustering is usually unsupervised and exploratory
- group observations
 - 'similar' observations are grouped together
 - 'dissimilar' observations are in different groups
- depends on definition of 'similarity' / distance



clustering

introduction

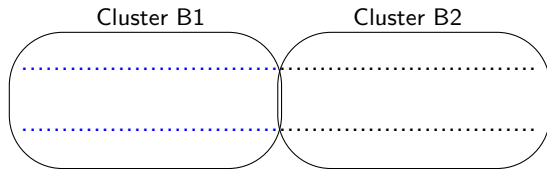
- clustering is usually unsupervised and exploratory
- group observations
 - 'similar' observations are grouped together
 - 'dissimilar' observations are in different groups
- depends on definition of 'similarity' / distance



clustering

introduction

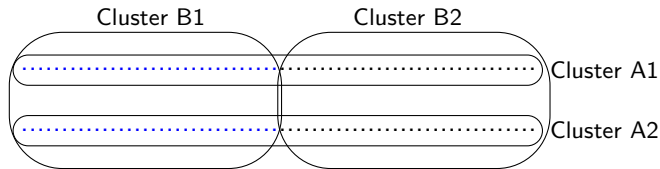
- clustering is usually unsupervised and exploratory
- group observations
 - 'similar' observations are grouped together
 - 'dissimilar' observations are in different groups
- depends on definition of 'similarity' / distance



clustering

introduction

- clustering is usually unsupervised and exploratory
- group observations
 - 'similar' observations are grouped together
 - 'dissimilar' observations are in different groups
- depends on definition of 'similarity' / distance



clustering

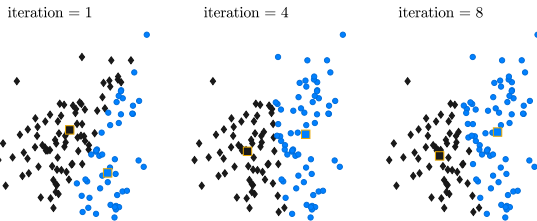
kMeans clustering

- 1 *Initialization*: randomly select K observations from the data set as initialization.
- 2 *Update*: compute the mean for each cluster.
- 3 *Assignment*: assign each observation to the cluster with the mean of the closest cluster.
- 4 *Iteration*: go to step 2 until the clusters converge.



clustering

kMeans clustering



distances

overview

■ *Euclidean Distance* (L2 Distance)

■ *Manhattan Distance* (L1 Distance)

■ *Cosine Similarity/Distance*

- range is from $[-1; 1]$ ($[0; 1]$ for non-negative input),
- not distance but similarity measure
- independent of vector length, only on angle

■ *Kullback-Leibler Divergence*

- not symmetric:
 $d_{KL}(\mathbf{v}_a, \mathbf{v}_b) \neq d_{KL}(\mathbf{v}_b, \mathbf{v}_a)$,
- designed to measure distance between probability distributions

$$d_{EU}(\mathbf{v}_a, \mathbf{v}_b) = \|\mathbf{v}_a - \mathbf{v}_b\|_2 = \sqrt{\sum_{j=0}^{\mathcal{T}-1} (v_a(j) - v_b(j))^2}.$$

distances

overview

- *Euclidean Distance* (L2 Distance)
- *Manhattan Distance* (L1 Distance)
- *Cosine Similarity/Distance*
 - range is from $[-1; 1]$ ($[0; 1]$ for non-negative input),
 - not distance but similarity measure
 - independent of vector length, only on angle
- *Kullback-Leibler Divergence*
 - not symmetric:
 $d_{KL}(\mathbf{v}_a, \mathbf{v}_b) \neq d_{KL}(\mathbf{v}_b, \mathbf{v}_a)$,
 - designed to measure distance between probability distributions

$$d_M(\mathbf{v}_a, \mathbf{v}_b) = \|\mathbf{v}_a - \mathbf{v}_b\|_1 = \sum_{j=0}^{J-1} |v_a(j) - v_b(j)|.$$

distances

overview

- *Euclidean Distance* (L2 Distance)
- *Manhattan Distance* (L1 Distance)
- *Cosine Similarity/Distance*
 - range is from $[-1; 1]$ ($[0; 1]$ for non-negative input),
 - not distance but similarity measure
 - independent of vector length, only on angle
- *Kullback-Leibler Divergence*
 - not symmetric:
 $d_{KL}(\mathbf{v}_a, \mathbf{v}_b) \neq d_{KL}(\mathbf{v}_b, \mathbf{v}_a)$,
 - designed to measure distance between probability distributions

$$s_C(\mathbf{v}_a, \mathbf{v}_b) = \frac{\sum_{j=0}^{\mathcal{J}-1} v_a(j) \cdot v_b(j)}{\sqrt{\sum_{j=0}^{\mathcal{J}-1} v_a(j)^2} \cdot \sqrt{\sum_{j=0}^{\mathcal{J}-1} v_b(j)^2}}.$$

$$d_C(\mathbf{v}_a, \mathbf{v}_b) = 1 - s_C(\mathbf{v}_a, \mathbf{v}_b).$$

distances

overview

- *Euclidean Distance* (L2 Distance)
- *Manhattan Distance* (L1 Distance)
- *Cosine Similarity/Distance*
 - range is from $[-1; 1]$ ($[0; 1]$ for non-negative input),
 - not distance but similarity measure
 - independent of vector length, only on angle
- *Kullback-Leibler Divergence*
 - not symmetric:
 $d_{KL}(\mathbf{v}_a, \mathbf{v}_b) \neq d_{KL}(\mathbf{v}_b, \mathbf{v}_a),$
 - designed to measure distance between probability distributions

$$d_{KL}(\mathbf{v}_a, \mathbf{v}_b) = \sum_{j=0}^{\mathcal{J}-1} v_a(j) \cdot \log \left(\frac{v_a(j)}{v_b(j)} \right).$$

summary

lecture content

■ regression

- model to estimate numeric labels from features
- linear regression assumes model is straight line

■ clustering

- 1 unsupervised grouping
- 2 feature space and distance measure determine result
- 3 number of clusters usually has to be known
- 4 kMeans is simple way of clustering

■ distances

- L1 and L2 are most common distances
- not all 'distances' are consistent. a real distance
 - ▶ cannot be negative
 - ▶ is symmetric

