Feature Selection and Engineering

SHALA2020.github.io

Learning objectives

- List advantages and disadvantages of feature selection and engineering
- List basic feature selection method types
- Write the formulae for a few feature selection metrics, e.g. t-test, AIC
- Write the steps for a few feature selection procedures, e.g. FS, BE
- Write the objective of LASSO and elastic-net
- Write the steps for PCA
- List some common features used for images, speech, and text

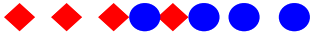
Why select features?

- Reduce overfitting
- Reduce confusion
- Reduce collinearity
- Reduce training time
- Simplify interpretation

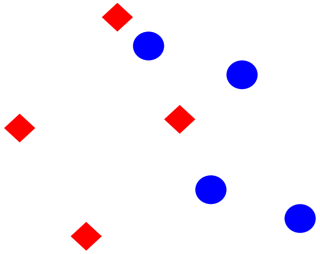
Disadvantages

- Inadvertently throw away useful information, because all selection methods have their own assumptions and biases

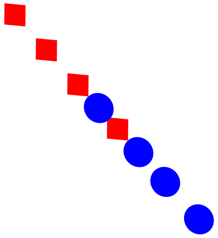
Compare two features



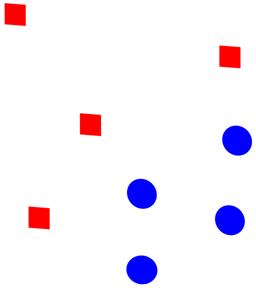
One of the
dimensions may
be useless



One of the
dimensions may
be useless

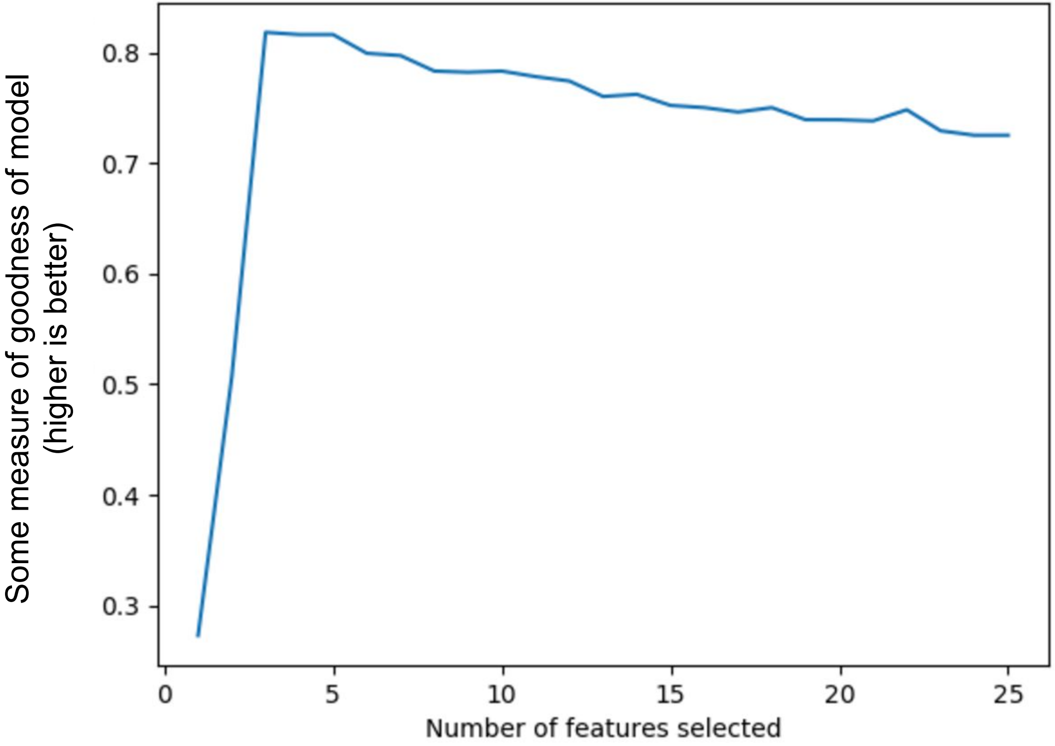


Rotating the
axes may be
useful



Rotating the
axes may be
useful

Usually, there is an optimal number (and subset) of features



Filtering

Basic idea: Remove “useless” or “redundant” features

- What makes a feature useful?
- What makes it non-redundant?

Algorithm:

- Loop through features x_i
  - Compute measure of utility and non-redundancy $m(x_i)$
- Sort features based on measure m
- Pick the top-k features

Filtering measures for regression

- Pearson correlation

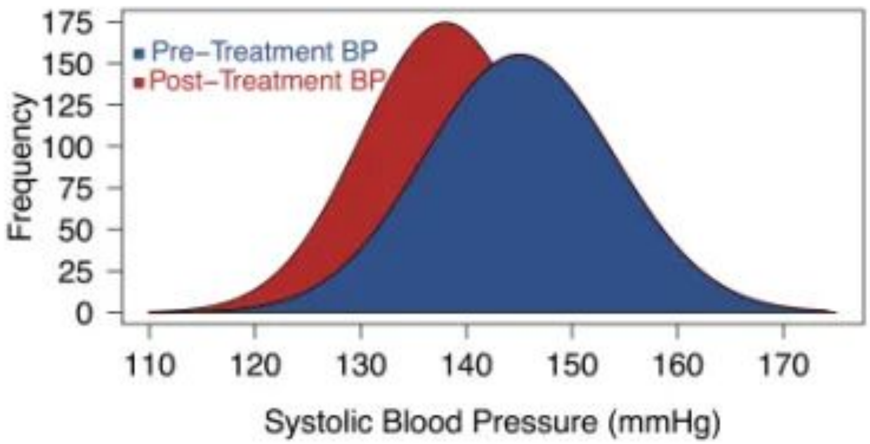
$$\frac{Cov(x,y)}{\sqrt{Var(x)}\sqrt{Var(y)}}$$

- Spearman correlation

$$\frac{Cov(rank(x),rank(y))}{\sqrt{Var(rank(x))}\sqrt{Var(rank(y))}}$$

Filtering measures for classification

- Hypothesis testing such as t-test, Wilcoxon rank test etc.



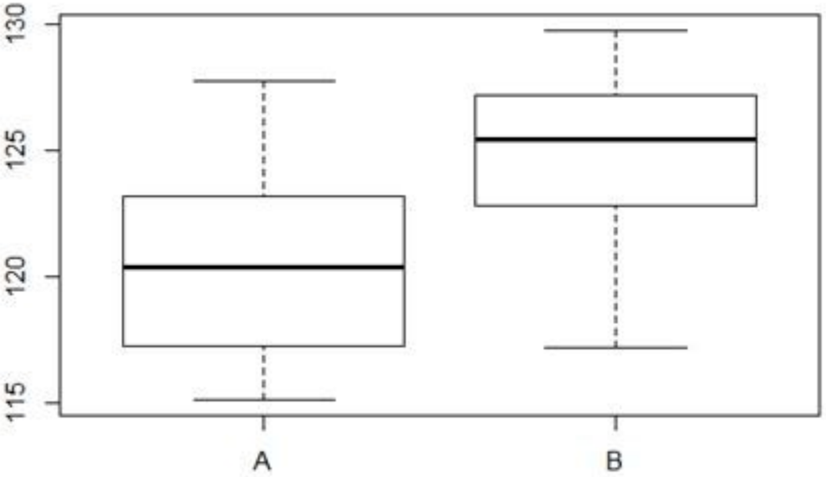
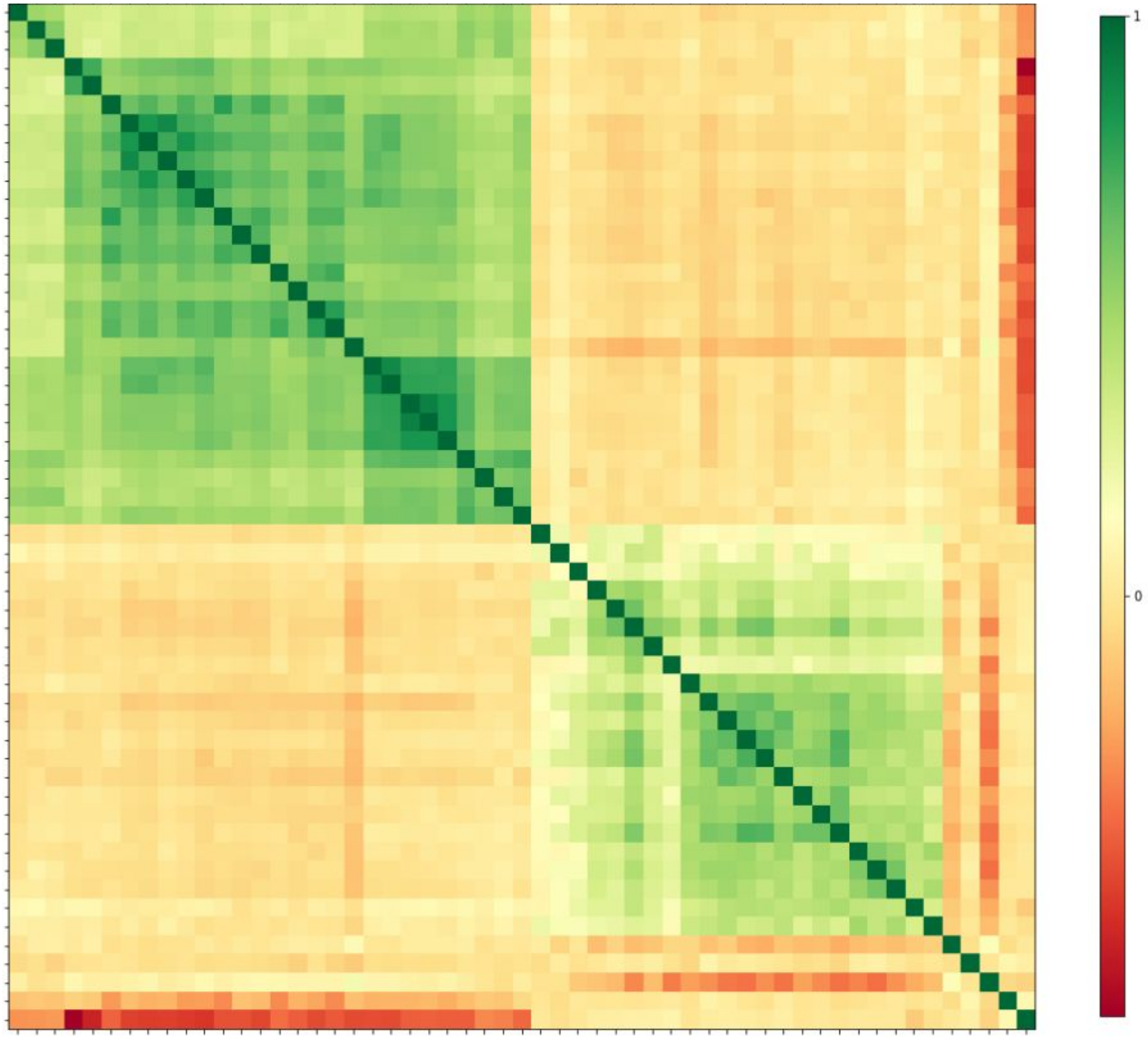


Image sources: https://www.r-bloggers.com/t-tests/, https://data.library.virginia.edu/the-wilcoxon-rank-sum-test/

Correlation-based clustering removes redundant features

One can choose a representative variable of each cluster

The selected variable can be based on interpretability



Wrapper methods

Basic idea:

- Add or remove features
- And measure model accuracy

Model selection criteria:

- K-fold CV
- AIC
- BIC

K-fold cross-validation

Held-out	Training

Result 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Result 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Result 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Result 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Result 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

Overall result

AIC and BIC

AIC - Akaike Information Criterion

- $2k - 2\ln(L)$; where k is number of parameters, L is likelihood

BIC - Bayesian Information Criterion

- $k\ln(n) - 2\ln(L)$; where k is number of parameters, n is number of samples, L is likelihood

Forward selection

- Start with no variables in set S and all variables in set A
- For i = 1 to d
  - for j = 1 to d-i+1
    - Compute a measure m of adding variable $x_{A(j)}$ from A to the model
    - Select the variable with the best measure
      - If the change in measure m meets some criteria
        - Remove it from A and put it in S
      - Else exit

Backward elimination (also RFE)

- Start with all variables in set S and no variables in set A
- For i = 1 to d
  - for j = 1 to d
    - Compute a measure m of removing variable $x_{S(j)}$ from S to the model
    - Select the variable with the best change in measure
      - If the change in measure m meets some criteria
        - Remove it from S and put it in A
      - Else exit

Forward selection and backward elimination compared

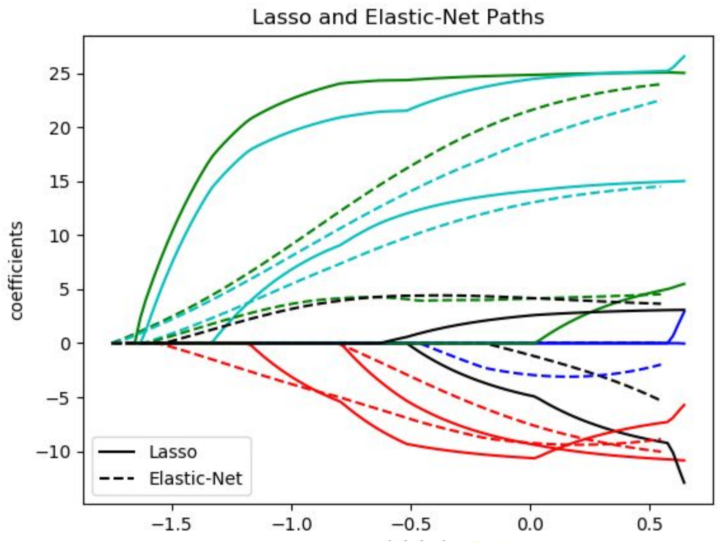
- Both are greedy methods
- In FS two variables individually may be uninformative and thus not considered, but together may be informative
- In BE we may start with a very complicated model initially itself

Regularization for feature selection

- LASSO (L1 penalty over weights)

$$L_p \text{ norm is } \left(\sum_i |w_i|^p \right)^{1/p}$$

- Elastic-net (L1+L2 penalty over weights)



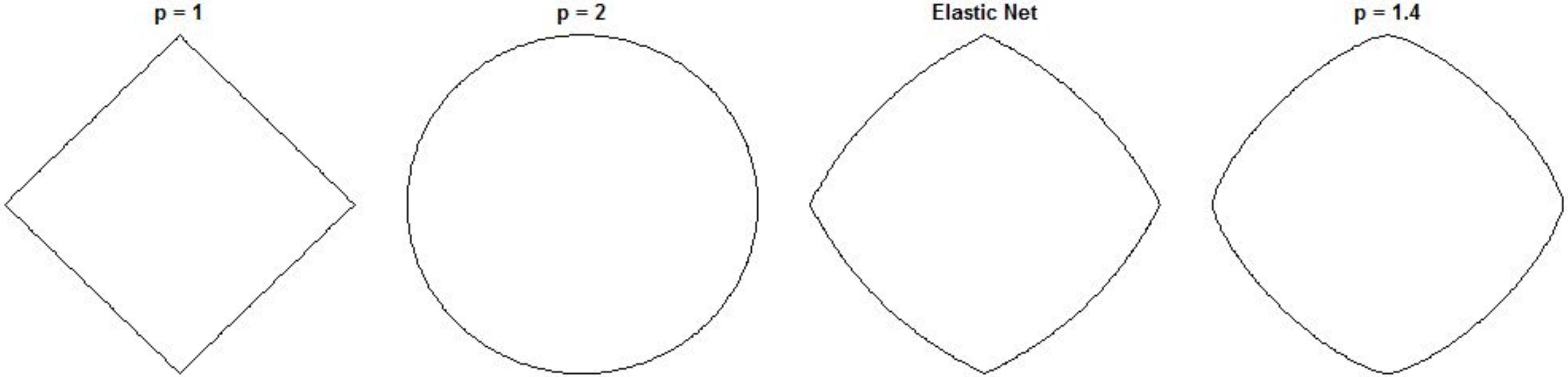


Image source: https://stats.stackexchange.com/questions/224531/bridge-penalty-vs-elastic-net-regularization; https://scikit-learn.org

Unsupervised feature reduction using PCA

Principal component analysis

- Start with *D*-dimensional data
- Compute covariance matrix Σ
- Eigen decompose $\Sigma = U \Lambda U^{\top}$
  - *U* contains eigenvectors (principal directions) and *Λ* is a diagonal matrix of eigenvalues
- Select *d*<*D* directions (eigenvalues) with the highest eigenvalues
- Project data to those *d* dim.s

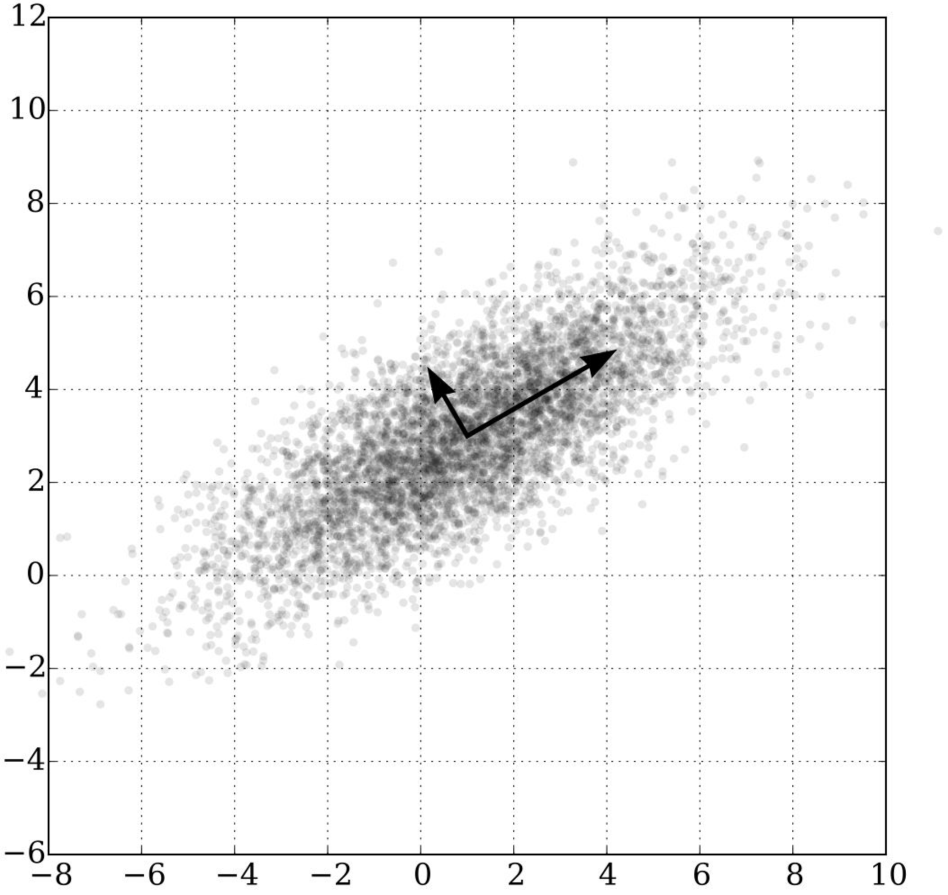


Image source: wikipedia.org

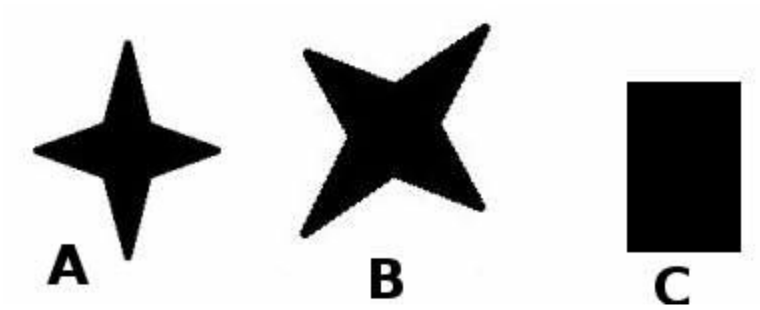
How to engineer features?

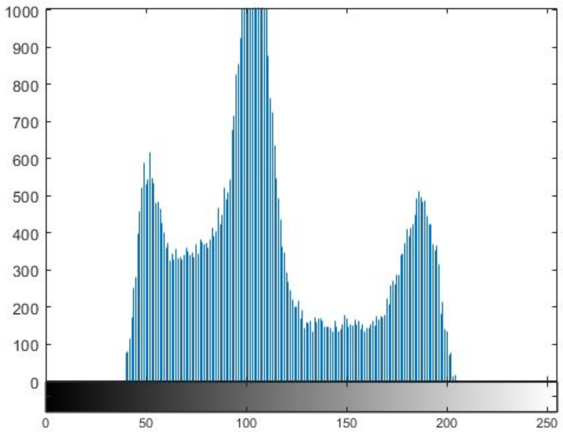
Features based on domain knowledge and statistics for:

- Images
- Audio
- Text

Features for images

- Intensity histogram
  - Its mean, median, mode, skew, kurtosis
- Color histogram
  - Bivariate histogram
- Hu invariant moments





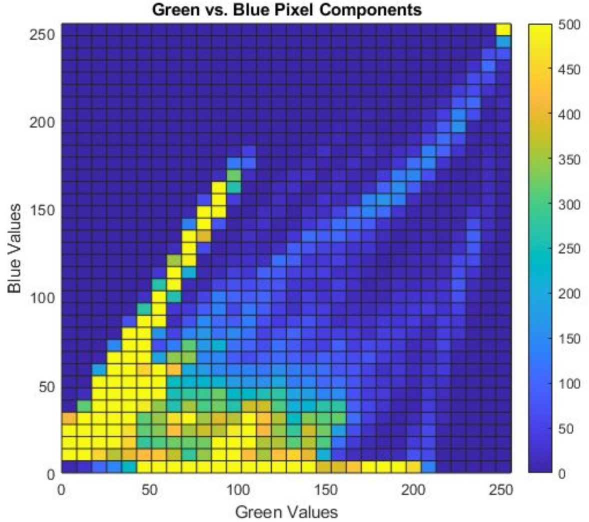
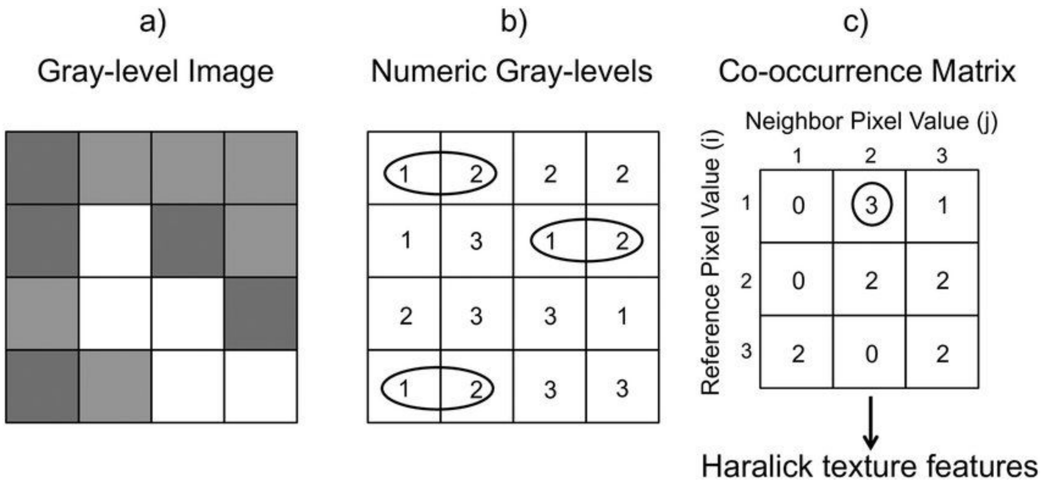


Image source: mathworks.com, opencv.org

Features for images

GLCM



Haar features

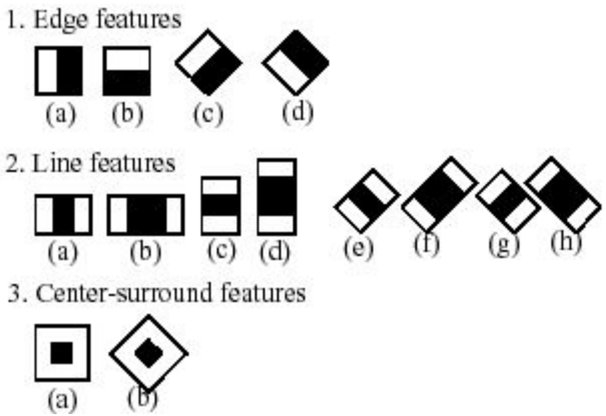


Image source: Do, Quyen et al. Texture analysis of magnetic resonance images of the human placenta throughout gestation: A feasibility study. PLOS ONE’19; opencv.org

Features for audio

MFCC features

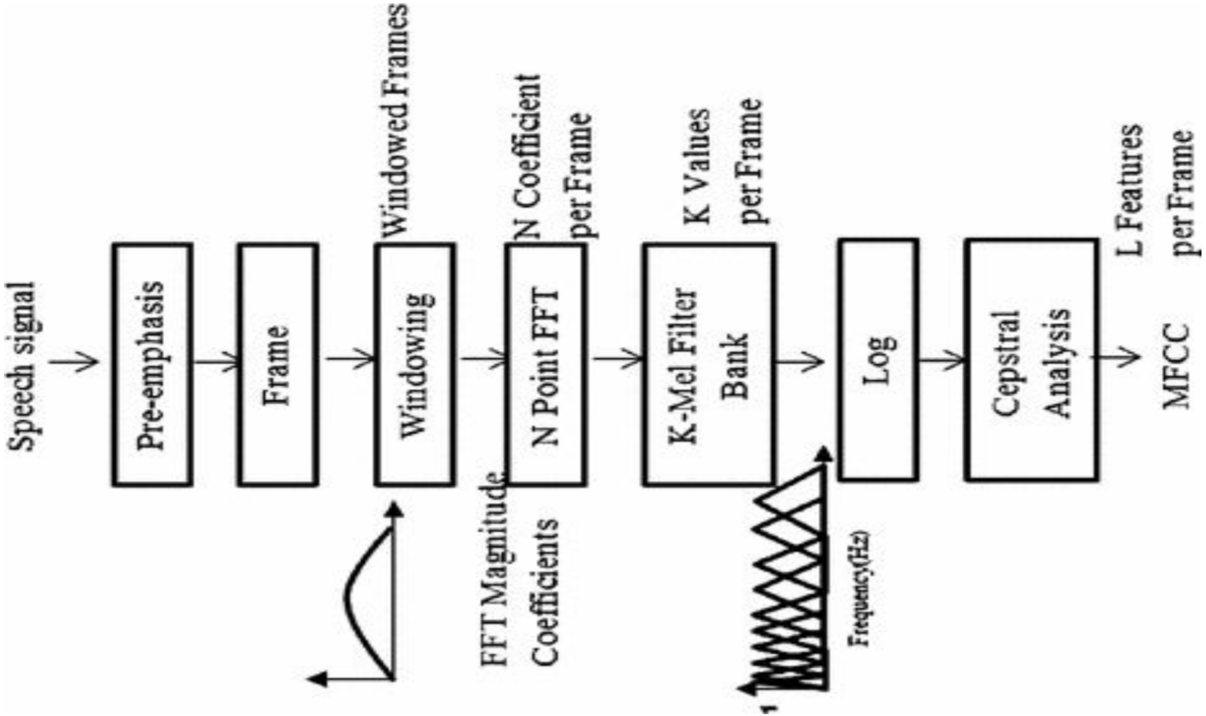


Image source: Srivastava S, Bhardwaj S, Kiran P. Gaussian Membership Function-Based Speaker Identification Using Score Level Fusion of MFCC and GFCC. InProceedings of the International Congress on Information and Communication Technology 2016 (pp. 283-291). Springer, Singapore.

Features for text

- Bag-of-words and Frequency of *n*-grams
  - "India posted a score of 256/8 in their allotted 50 overs in the third and deciding ODI of the series. Virat Kohli was the top-scorer for men in blue with a classy 71, while Adil Rashid and David Willey picked up three wickets each"
  - Counts
    - $\begin{array}{c} India \\ Posted \\ Score \end{array} \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$
  - The counts can be normalized
  - The words can be standardized
    - Score
    - Scorer
  - What about uninformative words?
- TF-IDF
  - Term frequency – inverse document frequency
  - TF $f_{t,d}$ is the count of term *t* in document *d*
    - Usually normalized in some sense
    - $\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$
  - IDF penalizes terms that occur often in all documents, e.g. "the"
    - $\mathrm{idf}(t,D) = \log \frac{|D|}{1+|\{d \in D : t \in d\}|}$
  - TF-IDF is $\mathrm{tf}(t,d) \times \mathrm{idf}(t,D)$
  - Form a vector of TF-IDF for various terms
    - Which terms?