FUTURE INSTITUTE OF ENGINEERING AND MANAGEMENT

CC – 148

UNDER

MAKAUT, WB

SPARSE MODELING & ESTIMATION

CONTINUOUS ASSESSMENT#2

MACHINE LEARNING

PEC-IT701D

Odd Semester

Academic Year: 2024-25

PRESENTED BY

POUSALI CHOWDHURY

14800221027

BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY

7TH SEMESTER

Sparse Modeling and Estimation

Sparse modeling is a collection of data analysis techniques that help understand and isolate the factors that have an impact on an occurrence.

Sparse modeling can provide feedback that explains the results and reasoning behind its solutions, which is an advantage over conventional AI methods.

Features :-

- Sparse data are features that have mostly zero values. This is different from features with missing data.
- A sparse matrix or sparse array is a matrix in which most of the elements are zero.
- But, if most of the elements of a matrix or array are nonzero then the matrix is considered as dense.

Examples :

Sparse features include vectors of one-hot-encoded words or counts of categorical data. But, features with dense data have predominantly non-zero values.

- The number of zero valued elements divided by the total number of elements is called sparsity of the matrix.

$$\begin{pmatrix} 1 & 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 4 & 5 & 0 \\ 0 & 0 & 0 & 8 & 6 \\ 7 & 0 & 9 & 0 & 0 \end{pmatrix} = M$$

In matrix M, there are 9 non-zero values and 16 zero values

$\therefore$ sparsity = 64 %

& density = 36 %

Example :-

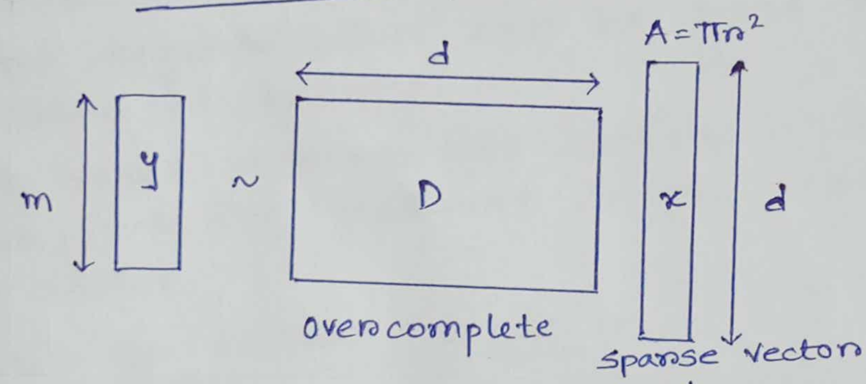
Given the observed value y,
the purpose is to find the factor x from
which the value y was obtained

$$\boxed{y = f(x)}$$

(Inverse problem)

It is focus on finding out why the value y
was obtained.

- If a model is define as sparse, it means that some
dependent variables are represented by 0 are not
used to predict. So, we can analyze the perfor-
mance of model from selected variable. It is
called model interpretability.



Sparse Representation

- Difference between Sparse data and missing data :-

When there is missing data, it means that many data
points are unknown. But, if the data is sparse, all
the data points are known, but most of them have
zero value.

To illustrate this, there are two types of features.
The feature with sparse data has known values (=0),
but the feature with missing data has unknown
values (=null). It is unknown what values should be

in the null-valued rows.

Row	Feature with Sparse Data	Feature with Missing Data
1	0	Null
2	0	2
3	1	Null
4	1	3

Methods to deal with sparse Features :-

(1) Removing features from the model :-

- Sparse features can introduce noise, which the model picks up and increase the memory needs of the model.
- To remedy this, the features can be dropped from the model.

Example : Rare words are removed from data mining models, or features with low variance are removed.

- But, Sparse features that have important signals should not be removed in this process.
- LASSO regularization can be used to decrease the number of features.
- Rule based methods like setting a variance threshold for including features in the model might also be useful.

(2) Make the features dense :-

- Principal Component Analysis (PCA) :

PCA methods can be used to project the features into the directions of the principal components and select from the most important components.

- Feature Hashing :

In feature hashing, sparse features can be binned into the desired number of output features using a hash function.

Care must be taken to choose a generous number of output features to prevent hash collisions.

(3) Using models that are robust to sparse features:

Some versions of machine learning models are robust towards sparse data and may be used instead of changing the dimensionality of the data.

Example: The entropy-weighted k-means algorithm is better suited to this problem than the regular k-means algorithm.

Uses of Sparse Modeling:

(1) Machine Learning:

Sparse modeling is widely used in machine learning for feature selection, dimensionality reduction, and creating interpretable models, such as in LASSO regression.

(2) Natural Language Processing:

In NLP, sparse modeling helps in text classification and topic modeling by representing documents with a small number of relevant words or topics.

(3) Signal Processing:

In signal Processing, sparse modeling is used for compressive sensing, where signals are reconstructed from a small number of measurements.

(4) Robust Statistical Modeling:

In statistics, sparse models are used to build models that are resistant to overfitting and can handle high-dimensional data where traditional models would struggle.

(5) Bioinformatics:

Identifying relevant genes in gene expression data for disease prediction.

(6) Econometrics:

Selecting key economic indicators while excluding irrelevant variables for more reliable models.

Fundamental Aspects of Sparse Modeling :-

(1) L1 Regularization (LASSO) :

- Promotes sparsity by shrinking some coefficients to 0.
- Commonly used for feature selection in regression models.

(2) L2 Regularization (Ridge Regression) :

- Shrinks coefficients but does not enforce exact sparsity.
- Often compared with LASSO for handling high-dimensional data.

(3) Bayesian Sparse Models :

- Utilizes sparse priors, such as the laplace distribution or spike-and-slab priors, to enforce sparsity in Bayesian models.
- Enables probabilistically sound sparse representations.

(4) Variational Inference for Sparse Modeling :

- Approximates complex posterior distributions in Bayesian sparse models.
- Used to efficiently compute sparse solutions in high-dimentional settings.

(5) Sparse Coding :

- Represents data as a sparse combination of basis functions.
- Widely used in signal processing and image processing.

(6) Sparse Matrix Computations :

Focuses on efficiently handling and computing with matrices that have a high number of zero elements.

Benefits of Sparse Modeling :

(1) Efficiency in Representation :
Sparse models use fewer coefficients to represent data, leading to more efficient storage and faster computations, especially when dealing with large datasets.

(2) Feature Selection :
Sparse modeling inherently performs feature selection by assigning zero weights to less important features, which can improve model interpretability and reduce overfitting.

(3) Robustness to Noise :
It can be more robust to noise in data, as they tend to ignore irrelevant or noisy features, improving the model's performance on noisy datasets.

(4) Interpretability :
Sparse models are easy to interpret, as they use fewer features, making it easier to understand the relationships in data.

(5) Improved Generalization :
By focusing on only the most relevant features, sparse models often generalize better to new, unseen data compared to models that can be use all available features.

Disadvantages of Sparse Modeling :

(1) Computational Complexity :
Finding the optimal sparse representation can be computationally expensive especially for large-scale problems or when the dictionary is large.

(2) Limited Expressiveness :
Sparse model migh not capture all the complex patterns in data, especially when the true underlying relationships are not sparse.

(3) Assumption of Sparsity :
Sparse modeling assumes that the underlying data can be well-represented by a sparse model, which may not be the case for all types of data.

(4) Sensitivity of Regularization Parameter (λ):

The choice of the regularization parameter is crucial. If not selected appropriately, the model may either be too sparse (underfitting) or not sparse enough (overfitting).

(5) Difficulty in Model Tuning:

Tuning sparse models, such as selecting the right level of sparsity or the best regularization parameter, can be challenging and often requires cross-validation or other techniques, which can be time-consuming.

Conclusion :-

Sparse modeling and estimation are fundamental techniques in efficiently representing data, especially in high-dimensional settings. These methods have broad applications across diverse domains, provide roburst solutions to complex problems. However, precise model tuning is crucial, as the choice of sparsity levels and regularization parameters significantly influences overall performance.

References :-

- "Pattern Recognization and Machine Learning" - by Christopher M. Bishop
- "Machine Learning : A Probabilistic Perspective" — by Kevin P. Murphy