

SPPU-BE-COMP-CONTENT - KSKA Git

ML

classmate

Date :

Page :

ASSIGNMENT-2

Q1

Reason: KNN is Lazy Learner (no training computation happens at prediction). For each new data point it calculates distances to all points in dataset.

Problem: For large datasets, this becomes computationally expensive in terms of both time & memory.

Conclusion: KNN is inefficient for large datasets where fast predictions are required.

Q2

→ Yes feature scaling is required for KNN.

→ Justification:

- KNN uses distance metrics to find neighbours.
- If features are on different scales the feature with larger values will dominate the distance calculation.

ex: If age $\in [0-100]$ & income $\in [0-100000]$
income will overshadow age in distance computation.

solution: apply feature scaling so all features contribute equally.

SPPU-BE-COMP-CONTENT - KSKA Git

Q3

- Hinge loss: is a loss function mainly used for Support Vector Machines (SVM)

- Formula:

$$L = \max(0, 1 - y \cdot g(x))$$

$y \rightarrow$ is the true label (-1 or +1) & $g(x)$ is the predicted score

- Intuition:

If prediction is correct & far from boundary \rightarrow Loss = 0

If prediction is wrong or too close to the decision boundary \rightarrow Positive Loss (Penalty)

- Purpose: encourages not just correct classification but also a margin of separation.

Q4

Kernel Trick: Mathematical technique in SVM's that allow computation in higher dimensional space without explicitly transforming the data.

How it works: Instead of mapping data into higher dimensions we use a kernel-function to compute inner products directly in that space.

SPPU-BE-COMP-CONTENT – KSKA Git

Usefulness:

- Makes SVM capable of handling non linear classification or regression.
- Avoids the high cost of explicitly computing coordinates in high dimensional space.
- Enables models to find complex decision boundaries efficiently.