**Machine Learning Final scope**

**Outliers** are data points that are significantly different from the rest of the dataset. This occur due to measurement errors or genuine extreme values. Outlier detection is crucial for improving model accuracy and robustness.

**Common techniques for handling outliers**

* **Remove outliers** – if an outlier is the results of data entry error, it should be removed.

**# Using Z-score to remove outliers**

z\_scores = np.abs(stats.zscore(df['value']))

df\_no\_outliers = df[(z\_scores < 3)]

* **Transform data** – use logarithmic transformation or robust scaling.

**# Log transformation**

df['log\_value'] = np.log1p(df['value'])

# Robust scaling

from sklearn.preprocessing import RobustScaler

scaler = RobustScaler()

df['robust\_scaled'] = scaler.fit\_transform(df[['value']])

* **Cap extreme values** – replace extreme values with predefined threshold e.g., 95th percentile.

**Bias-variance tradeoff** – is a fundamental concept that help explain the balance between model complexity and generalization.

**Bias** – error is introduced when the model is too simplistic to capture the underlying structure of the data, leading to Underfitting that causes systematically incorrect prediction.

**Variance** – this error is due to the model being too sensitive to small fluctuation in the training data. This model overfits, capturing noise instead of meaningful patterns.

**Impact of Bias and Variance on model performance.**

If model has high **bias**, it performs poorly even on training data, indicating Underfitting.

**Ways to prevent Bias**

* Add more features.
* Increase model complexity.
* Feature engineering.
* Reduce regularization.

If model has high **variance**, it performs well on training data but poorly on unseen data, indicating overfitting.

**Ways to prevent variance**

* Simplify the model (reduce polynomial degree).
* Increase the amount of training data.
* Use regularization techniques like L1/L2 penalties to constrain the model complexity.
* Cross-validation.

**SECOND SESSION**

Polynomial curve

K-Nearest Neighbors

K-Means Clustering

Logistic Regression

Support Vector Machines

Decision Tree Regressor/Classifier

Random Forest

**THIRD SESSION**

ANN

ROC curve

Activation functions

Regularization techniques