CS506 Notes

Lecture 10: Classification

- Predict a class label using input features (predictors).
- Success depends on correlation between predictors and target class.
- Imperfect prediction is expected due to noise or inadequate features.
- **Correlation:**
  - Use Pearson for linear data; Spearman for ordinal/nonlinear.
- **Data Types:**
  - Nominal: no inherent order (e.g., color).
  - Ordinal: ordered, but gaps aren't meaningful (e.g., ratings).
- **Model Evaluation:**
  - Use separate training and testing sets to avoid overfitting.
- **K-Nearest Neighbors (KNN):**
  - Predict using majority class of nearest neighbors.
  - Pros: simple, interpretable.
  - Cons: slow for large datasets; suffers in high-dimensional space.

Lecture 11: Decision Trees

- Predict class via yes/no paths down a tree.
- **Hunt's Algorithm:** Recursively split data to create pure subsets.
- **Splits:**
  - Binary (e.g., age ¿ 30).
  - Multi-way (e.g., weather = sunny/rainy/overcast).

- **GINI Index:** Measures impurity of a node.
- **Overfitting:**
  - Avoid by early stopping or pruning.

Lecture 12: Model Evaluation

- **Confusion Matrix Metrics:**

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$
$$\text{Precision} = \frac{TP}{TP + FP}$$
$$\text{Recall} = \frac{TP}{TP + FN}$$
$$\text{F1 Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Validation Methods:**
  - Holdout, K-Fold Cross Validation, Leave-One-Out (LOO)
- **Ensemble Methods:**
  - Combine multiple models to reduce error.
  - **Bagging:** Build models on bootstrap samples (e.g., Random Forest).
  - **Boosting:** Sequentially train models to correct previous errors.

Lecture 13: Support Vector Machines (SVM)

- **Goal:** Find the widest possible margin separating classes.
- **Decision Boundary:** $w^T x + b = 0$
- **Regularization Parameter (C):**
  - $C > 1$: Narrow margin, fewer errors, risk of overfitting.
  - $C < 1$: Wider margin, tolerant of errors, better generalization.
- **Soft Margin:** Allows some misclassifications.
- **Kernel Trick:** Transforms data to higher dimensions for linear separation using kernel functions.

Lecture 14: Recommender Systems

- **Challenges:** Scale, cold start, sparse data.
- **Methods:**
  - **Neighborhood-Based:** Recommend based on similar users/items.
  - **Content-Based Filtering:** Use item features to recommend similar items.
  - **Collaborative Filtering:** Matrix factorization to discover latent user/item features.

Lecture 15: Linear Regression

- **Goal:** Fit a linear model $y = X\beta$ to predict target values.
- **Assumptions:**
  - Linearity, independence, and normality of residuals.
- **Methods:**
  - **Least Squares:** Minimize $\sum(y_i - \hat{y}_i)^2$.
  - **Maximum Likelihood:** Maximize $P(Y \mid h)$ assuming Gaussian noise.