KENT STATE UNIVERSITY

# Assignment #4

**Total Points:** 200

**Office location**: CAED 210H

## Instructions

**Collaboration Policy:** For "Individual assignments," Collaboration is expected within the limits of discussing concepts and problems. However, each student must produce his/her own solution to the problems. For "Group assignments," each student should have specific contributions to the homework –by default, the instructor assumes equal contributions; if any disagreement between team members about the contributions of individuals, please talk to the instructor to initiate a peer evaluation process.

**ChatGPT Policy**: In general, please be transparent if you use ChatGPT and highlight the parts of the homework generated by ChatGPT. When you choose to use ChatGPT to provide some answers, please *1) use an online document to save the ChatGPT sessions that helped you produce the answer*; 2*) critically review the answers generated by ChatGPT, highlight the parts that you found that ChatGPT's answer needs improvements*.

**General Expectations and Requirements for Homework:**

- Please use considerable and uniform font sizes throughout the document to maintain consistency of the document. You may choose to highlight subheadings with either a bold or underlined feature.
- Please use bullet points wherever possible to make the answers clear and easy to follow by an educated reader.
- Please do not forget to reference additional data, hyperlinks or literature used as evidence or background information to support your claims and solutions in the document. Please list those references below your answer or at the end of the document.
- Please refer to the textbook and provide descriptive answers wherever possible.
- Please communicate with the instructor to clarify questions about the homework description BEFORE the submission; after the homework submission deadline, the students are responsible for the point losses due to different ways of interpreting the homework requirements.

All homework submissions should be submitted electronically on Canvas.

**Building Permit Analysis & ML in Construction**

Student Name:
Date:
Course/Instructor:

In this assignment, you will apply end-to-end machine learning methods to a real-world construction dataset. The goal is to gain experience with:

- Data Collection & Cleaning (handling missing values, formatting).
- Exploratory Data Analysis (EDA) and feature engineering.
- Model Development (train/test split, avoiding data leakage).
- Model Evaluation (metrics, confusion matrix or error metrics).
- Advanced ML Techniques (e.g., random forest, XGBoost, or hyperparameter tuning).

**Dataset**
We will use the **Seattle Building Permits** dataset from Kaggle:

- **Link**: Seattle Building Permits | Kaggle
- **Description**: Contains details on building permit applications in the City of Seattle. Fields may include:
    - **Permit Class** (Residential, Commercial, etc.)
    - **Permit Type** (Major, Minor)
    - **Application/Issue/Expiration Dates**
    - **Project Value** (estimated cost)
    - **Status** (e.g., Approved, Cancelled, Completed)

**Deliverables**

- A **Jupyter Notebook** (.ipynb) documenting all steps:
    1. **Data cleaning** and **EDA**.
    2. **Feature engineering** and **model building**.
    3. **Model evaluation** and **analysis of results**.
    4. **Discussion** of data leakage avoidance and insights gained.
- A **short write-up** (1–2 pages or a well-documented notebook section) summarizing:
    - Your **methodology**.
    - Key **findings** (interesting EDA insights, model performance).
    - **Challenges** faced (missing data, outliers, etc.).
    - **Recommendations** (for future improvements or additional features).

**Hints for Detailed Steps**

**1. Data Acquisition & Description**

1) **Download** the Seattle Building Permits dataset (https://www.kaggle.com/datasets/city-of-seattle/seattle-building-permits)
2) **Load** the CSV file into a Pandas DataFrame.
3) Provide a **brief description** of the dataset:
    - Which columns are available?
    - What do they represent?
    - How many rows and columns?
    - Are there any duplicated rows?

- Note: If you prefer a different city's building permit data or another construction-related open dataset, that's acceptable. Adapt the tasks accordingly.

**2. Data Cleaning & Preprocessing**

1) **Inspect missing values** and decide how to handle them. For example:
    - Drop columns or rows with excessive missingness.
    - Impute with mean/median if appropriate.
    - Use domain knowledge if certain fields are critical (e.g., a project "Value" cannot be 0 if it's a new construction).
2) **Check for inconsistencies or outliers**:
    - Project values that are unrealistically large (billions) or negative.
    - Permit statuses that don't align with the project timeline.
3) **Feature Selection/Engineering**:
    - Create new features (e.g., *Permit Duration* = Date Issued – Date Applied).
    - Convert date fields to a usable numeric or timestamp format.
    - Encode categorical columns (Permit Class, Status, etc.) with one-hot encoding or label encoding as appropriate.

**Goal**: Produce a **clean, well-structured** dataset suitable for modeling, with potential new features that capture domain insights.

**3. Exploratory Data Analysis (EDA)**

Use **charts and statistics** to understand the distribution and relationships in your data:

1) **Distribution plots/histograms**: For numeric columns (Project Value, Duration).
2) **Bar charts**: Summaries by Permit Class (residential vs. commercial).
3) **Correlation heatmap**: Identify relationships between numeric features (e.g., cost vs. duration).
4) **Potential classification/regression target**:
    - **Regression Example**: Predict the total Project Value (cost) from other permit attributes.
    - **Classification Example**: Predict if a project is likely to be *Cancelled* vs. *Completed* (or *Issued* vs. *Not Issued*).

Note: Your target variable depends on data availability. If the "Status" column is well-defined and has distinct categories, consider classification. If the "Project Value" or "Duration" is more relevant, do regression.

**4. Model Definition & Splitting Strategy**

1) **Choose a target** (e.g., classification: Status, or regression: Project_Value).
2) **Split** your data into training and test sets (e.g., 80% / 20%).
    - If your data has a temporal component (like application date), consider a **time-based split** to avoid leakage from future data.
    - Alternatively, use a standard train_test_split if time-based ordering isn't crucial.
3) **Data Leakage Considerations**:
    - Ensure any transformations (e.g., scaling, feature engineering that uses aggregated stats) are done **after** the split using training data only.
    - Verify that no "future" fields (like final permit outcome date) are included in training if you're trying to predict that outcome.

**5. Build & Evaluate Models**

1) **Basic Model**: Start with a simple approach (e.g., **Linear Regression** for cost, or **Logistic Regression** for permit status).
2) **Advanced Models**: Try **Random Forest** or **XGBoost**:
    - For classification: measure *accuracy*, *precision*, *recall*, and *F1-score*.
    - For regression: measure *MAE*, *RMSE*, and $R^2$.
3) **Hyperparameter Tuning** (optional but encouraged):
    - Use GridSearchCV or RandomizedSearchCV for advanced models.
    - Document your parameter search ranges and final chosen parameters.
4) **Performance Comparison**:
    - Compare baseline vs. advanced vs. tuned model performance.
    - Create a small table or summary of metrics.

**6. Results & Analysis**

1) Display **evaluation metrics** for each model.
2) Discuss which model performed best and **possible reasons** (e.g., more robust to outliers, better at handling complex relationships).
3) If classification, **show a confusion matrix** or ROC curve. For regression, maybe a **scatter plot** of predictions vs. actual values.

**7. Final Write-Up / Reflection**

Provide a short summary (1–2 pages or a markdown section in your notebook) covering:

1) **Data Observations**:
    - Key insights from EDA (e.g., main drivers for project cost or statuses).
2) **Model Insights**:
    - Which model performed best and why you think so.
3) **Data Leakage Precautions**:
    - How you avoided leakage (splitting first, no future data, etc.).
4) **Limitations & Future Work**:
    - Data quality issues, potential domain knowledge you'd add in a real project.
    - Additional advanced methods.