Building a Wine Quality Prediction Model: A Machine Learning Project Walkthrough

In this lecture, the steps involved in a typical machine learning project are detailed using the example of building a model to predict wine quality based on its physiochemical properties. The goal is to replace an expensive quality sensor with a machine learning model.

Problem Framing and Design Considerations

The first crucial step is properly framing the problem. This involves defining the performance measure, listing and checking assumptions, and specifying the input and output. Key considerations include:

- Identifying the type of machine learning problem (supervised, unsupervised, reinforcement).
- Determining the nature of the output (single or multiple outputs).
- Understanding the business objective and how the model will be used.
- Establishing a baseline using the current solution (e.g., human labor).

> ***Important Note:*** *Strong collaboration with domain experts, product managers, and engineering teams is essential for successful project execution.*

Data Acquisition, Exploration, and Preprocessing

The next step involves accessing and exploring the data. In this example, data is downloaded from the UCI Machine Learning Repository (CSV format). A function for downloading and extracting data is recommended. Data exploration includes examining features (e.g., fixed acidity, volatile acidity, etc.) and labels (wine quality). The dataset used is relatively small (1599 entries).

Data Sampling and Splitting

Stratified sampling is employed to ensure the test set distribution matches the overall data distribution. The `StratifiedShuffleSplit` class from Scikit-learn is used for this purpose. The `train_test_split` function from `sklearn.model_selection` is used for splitting the data into training and testing sets. This function allows for random sampling with a specified random state, test size, and shuffling option.

```
from sklearn.model_selection import train_test_split, StratifiedShuffleSplit
# ... code to use these functions ...
```

Assumptions and Review

It's crucial to list and review assumptions made during problem-solving. These assumptions should be reviewed with domain experts and other relevant teams *before* coding begins. This ensures alignment and avoids potential issues later in the project. Questions to consider include whether continuous learning or periodic updates are needed, and whether a batch or online learning approach is more suitable.

Prof. Ashish emphasizes the iterative nature of machine learning projects, highlighting the importance of careful problem definition, data understanding, and collaboration to build effective and impactful models.