项目（P1）模板：钢铁生产数据分析

项目概述

本项目专注于分析钢铁生产数据，以构建用于质量控制和工艺优化的预测模型。你将实现一个完整的机器学习流水线，从数据预处理到模型对比。你的目标是预测一个描述钢材质量的因子（“output”）。

目标

你将：

- 实现全面的数据预处理技术
- 训练并比较多种回归模型
- 使用统计指标评估模型性能
- 创建专业的可视化与报告

数据集

下载链接：https://cloud.cps.unileoben.ac.at/index.php/s/xtsCHycaSiHsqHT

实现指南

阶段 1：环境搭建与数据加载

1.1 创建项目结构

```
mkdir steel_production_analysis
cd steel_production_analysis
mkdir scripts data results figures
```

1.2 所需包

```
pandas>=1.5.0
numpy>=1.21.0
matplotlib>=3.5.0
seaborn>=0.11.0
scikit-learn>=1.0.0
jupyter>=1.0.0
```

1.3 数据加载

创建 `scripts/01_data_loading.py`：

```
import pandas as pd
import numpy as np

def load_steel_data(file_path):
    """
    Load steel production dataset
    Returns: pandas DataFrame
    """
    # Data loading logic
    pass
```

阶段 2：数据预处理

2.1 数据清洗

创建 `scripts/02_data_preprocessing.py`：

需要实现的任务：

- 移除重复条目
- 处理缺失值（均值 / 中位数插补）
- 使用 IQR 方法检测并处理异常值
- 将分类变量转换为数值变量
- 检查数据一致性

示例：

```
def remove_duplicates(df):
    """Remove duplicate rows from dataset"""
    pass

def handle_missing_values(df):
    """Identify and impute missing values"""
    pass

def detect_outliers(df, columns):
    """Detect outliers using IQR method"""
    pass

def encode_categorical_variables(df):
    """Convert categorical columns to numerical"""
    pass
```

2.2 探索性数据分析

创建 `scripts/03_eda.py`：

需要创建的可视化：

- 相关矩阵热力图
- 所有特征的直方图

- 用于异常值检测的箱线图
- 特征关系的成对图（pair plots）
- 目标变量分布

示例：

```
def plot_correlation_matrix(df):
    """Create correlation heatmap"""
    pass

def plot_feature_distributions(df):
    """Plot histograms for all features"""
    pass

def plot_target_distribution(target):
    """Visualize target variable distribution"""
    pass
```

2.3 数据划分与归一化

- 对数据进行归一化
- 将数据划分为训练集、测试集和验证集

示例：

```
def split_and_normalize_data(df, target_column):
    """
    Split data into train/validation/test sets
    Normalize features using StandardScaler
    """
    pass
```

阶段 3：模型训练

3.1 实现多种回归模型

创建 `scripts/04_model_training.py`：

需要实现的模型：

- 随机森林回归（Random Forest Regressor）
- 支持向量机回归（Support Vector Machine, SVR）
- 多层感知机（Multi-Layer Perceptron, MLP）
- 高斯过程回归（Gaussian Process Regressor）
- （可选）LSTM 网络

```
def train_random_forest(X_train, y_train):
    """Train Random Forest model"""
    pass
```

```python
def train_svm(X_train, y_train):
    """Train Support Vector Machine model"""
    pass

def train_mlp(X_train, y_train):
    """Train Neural Network model"""
    pass

def train_gaussian_process(X_train, y_train):
    """Train Gaussian Process model"""
    pass
```

3.2 模型评估

- 对已训练模型进行评估。示例：

```python
def evaluate_model(model, X_test, y_test):
    """
    Evaluate model performance
    Returns: RMSE, MAE, R² scores
    """
    pass
```

阶段 4：结果分析与对比

4.1 性能指标计算

创建 `scripts/05_results_analysis.py`：

需要计算的指标：

- 均方根误差（Root Mean Square Error, RMSE）
- 平均绝对误差（Mean Absolute Error, MAE）
- 决定系数（R-squared, R^2）
- 训练时间
- 推理时间（预测时间）

```python
def calculate_metrics(y_true, y_pred):
    """Calculate performance metrics"""
    pass

def create_performance_table(results):
    """Create comparative performance table"""
    pass
```

4.2 可视化

可生成的示例图：

- 带误差条的柱状图进行模型对比
- 预测值与真实值的散点图
- 残差图
- 学习曲线

```
def plot_model_comparison(results):
    """Create bar plots with error bars"""
    pass

def plot_predictions_vs_actual(y_true, y_pred, model_name):
    """Scatter plot of predictions vs actual values"""
    pass

def plot_residuals(y_true, y_pred, model_name):
    """Plot residual analysis"""
    pass
```

阶段 5：最终报告

你的最终报告应使用课堂上提供的项目提交模板，并导出为 PDF。你可以使用 Overleaf 来撰写：https://www.overleaf.com/

5.1 撰写完整报告

需要包含的章节：

1. **Abstract（摘要）**：项目概要
2. **Introduction（引言）**：项目概览和目标
3. **Data Description（数据描述）**：数据集特征和预处理步骤
4. **Methodology（方法）**：所实现的模型及训练方法
5. **Results（结果）**：性能指标和可视化结果
6. **Discussion（讨论）**：模型对比与洞察
7. **Conclusion（结论）**：关键发现与建议

5.2 代码结构

```
steel_production_analysis/
|
├── data/
|   └── steel_production_data.csv
|
├── scripts/
|   ├── 01_data_loading.py
|   ├── 02_data_preprocessing.py
|   ├── 03_eda.py
```

```
|   ├── 04_model_training.py
|   └── 05_results_analysis.py
|
├── results/
|   ├── performance_metrics.csv
|   └── model_predictions/
|
├── figures/
|   └── ...
|
└── P1-doc.pdf
```

Good luck!