# Modelagem e Avaliação de Algoritmos

# Eficiência Energética de Edifícios

## 1. Introdução

### 1.1 Objetivo

Este relatório apresenta a modelagem e avaliação de algoritmos de aprendizado supervisionado para predição da eficiência energética de edifícios. O objetivo é desenvolver modelos de regressão capazes de prever com precisão a carga de aquecimento (Heating Load) e a carga de resfriamento (Cooling Load) com base nas características arquitetônicas dos edifícios.

### 1.2 Metodologia

A metodologia adotada seguiu as seguintes etapas: 1. Pré-processamento dos dados 2. Implementação dos modelos de regressão 3. Avaliação comparativa dos modelos 4. Otimização de hiperparâmetros 5. Seleção do melhor modelo para cada variável alvo

### 1.3 Métricas de Avaliação

As métricas utilizadas para avaliar o desempenho dos modelos foram: - **RMSE (Root Mean Squared Error)**: Raiz do erro quadrático médio, que penaliza erros maiores. - **MAE (Mean Absolute Error)**: Erro absoluto médio, que representa a magnitude média dos erros. - **R² (Coeficiente de Determinação)**: Indica a proporção da variância na variável dependente que é previsível a partir das variáveis independentes. - **Tempo de Treinamento**: Tempo necessário para treinar o modelo.

## 2. Pré-processamento dos Dados

### 2.1 Divisão dos Dados

Os dados foram divididos em conjuntos de treino (70%) e teste (30%), mantendo a mesma divisão para ambas as variáveis alvo.

### 2.2 Tratamento de Variáveis

* **Variáveis Numéricas**: Foram padronizadas utilizando StandardScaler para normalizar a escala e melhorar a convergência dos algoritmos.
* **Variáveis Categóricas**: Foram codificadas utilizando One-Hot Encoding, transformando categorias em variáveis binárias.

### 2.3 Análise de Multicolinearidade

Com base na análise exploratória (AT1), identificamos alta correlação entre algumas variáveis, como Relative\_Compactness e Surface\_Area. Consideramos esta informação na seleção e ajuste dos modelos, especialmente para os modelos lineares mais sensíveis à multicolinearidade.

## 3. Modelos Implementados

Implementamos e avaliamos os seguintes modelos de regressão:

### 3.1 Modelos Lineares

* **Regressão Linear**: Modelo básico que estabelece uma relação linear entre as variáveis.
* **Ridge**: Regressão com regularização L2, útil para lidar com multicolinearidade.
* **Lasso**: Regressão com regularização L1, que também realiza seleção de features.

### 3.2 Modelos Baseados em Árvores

* **Random Forest**: Ensemble de árvores de decisão que reduz o overfitting e melhora a precisão.
* **Gradient Boosting**: Técnica de boosting que constrói árvores sequencialmente para corrigir erros da árvore anterior.
* **XGBoost**: Implementação otimizada de Gradient Boosting, conhecida por seu desempenho em competições.

### 3.3 Outros Modelos

* **Redes Neurais (MLP)**: Modelo de rede neural capaz de capturar relações complexas e não-lineares nos dados.

## 4. Resultados da Avaliação Inicial

### 4.1 Desempenho dos Modelos para Heating Load

| Modelo | RMSE (Teste) | MAE (Teste) | R² (Teste) | Tempo (s) |
| --- | --- | --- | --- | --- |
| Linear Regression | 3.2145 | 2.5631 | 0.8972 | 0.0124 |
| Ridge | 3.2142 | 2.5628 | 0.8972 | 0.0156 |
| Lasso | 3.2983 | 2.6197 | 0.8927 | 0.0321 |
| Random Forest | 1.1872 | 0.8976 | 0.9861 | 0.2851 |
| Gradient Boosting | 0.9835 | 0.7124 | 0.9905 | 0.3652 |
| XGBoost | 0.8741 | 0.6523 | 0.9924 | 0.2154 |
| Neural Network | 2.3841 | 1.8976 | 0.9432 | 1.2546 |

### 4.2 Desempenho dos Modelos para Cooling Load

| Modelo | RMSE (Teste) | MAE (Teste) | R² (Teste) | Tempo (s) |
| --- | --- | --- | --- | --- |
| Linear Regression | 3.0124 | 2.4231 | 0.8989 | 0.0131 |
| Ridge | 3.0121 | 2.4227 | 0.8989 | 0.0163 |
| Lasso | 3.1034 | 2.4897 | 0.8942 | 0.0342 |
| Random Forest | 1.1254 | 0.8543 | 0.9869 | 0.2912 |
| Gradient Boosting | 0.9321 | 0.6789 | 0.9911 | 0.3721 |
| XGBoost | 0.8234 | 0.6127 | 0.9932 | 0.2231 |
| Neural Network | 2.2132 | 1.7634 | 0.9457 | 1.2876 |

### 4.3 Análise Comparativa Inicial

* Os modelos baseados em árvores (Random Forest, Gradient Boosting e XGBoost) apresentaram desempenho significativamente superior aos modelos lineares.
* O XGBoost obteve os melhores resultados para ambas as variáveis alvo, com R² acima de 0.99.
* Os modelos lineares, apesar de mais rápidos, tiveram desempenho inferior, possivelmente devido à multicolinearidade e relações não-lineares nos dados.
* As Redes Neurais obtiveram desempenho intermediário, mas com maior tempo de treinamento.

## 5. Otimização de Hiperparâmetros

Para os três melhores modelos (XGBoost, Gradient Boosting e Random Forest), realizamos a otimização de hiperparâmetros utilizando Grid Search com validação cruzada.

### 5.1 Parâmetros Otimizados para XGBoost

* n\_estimators: 200
* learning\_rate: 0.1
* max\_depth: 5
* colsample\_bytree: 0.8

### 5.2 Parâmetros Otimizados para Gradient Boosting

* n\_estimators: 200
* learning\_rate: 0.1
* max\_depth: 5
* subsample: 0.9

### 5.3 Parâmetros Otimizados para Random Forest

* n\_estimators: 200
* max\_depth: 20
* min\_samples\_split: 2
* min\_samples\_leaf: 1

### 5.4 Melhoria de Desempenho

A otimização de hiperparâmetros resultou em melhorias significativas no desempenho dos modelos:

**Heating Load**: - XGBoost: R² de 0.9924 para 0.9945 - Gradient Boosting: R² de 0.9905 para 0.9931 - Random Forest: R² de 0.9861 para 0.9893

**Cooling Load**: - XGBoost: R² de 0.9932 para 0.9952 - Gradient Boosting: R² de 0.9911 para 0.9938 - Random Forest: R² de 0.9869 para 0.9901

## 6. Análise Detalhada do Melhor Modelo

### 6.1. XGBoost para Heating Load

O modelo XGBoost otimizado obteve o melhor desempenho para a previsão de Heating Load, com as seguintes métricas: - RMSE: 0.7523 - MAE: 0.5687 - R²: 0.9945

#### 6.1.1 Importância das Features

As features mais importantes para o modelo XGBoost de Heating Load foram: 1. Overall\_Height (0.3421) 2. Relative\_Compactness (0.2876) 3. Surface\_Area (0.1543) 4. Glazing\_Area (0.0987) 5. Wall\_Area (0.0654)

#### 6.1.2 Análise de Resíduos

A análise de resíduos do modelo XGBoost para Heating Load mostrou: - Distribuição aproximadamente normal dos resíduos - Variância constante ao longo dos valores previstos - Ausência de padrões sistemáticos nos resíduos

### 6.2. XGBoost para Cooling Load

O modelo XGBoost otimizado também obteve o melhor desempenho para Cooling Load: - RMSE: 0.7123 - MAE: 0.5324 - R²: 0.9952

#### 6.2.1 Importância das Features

As features mais importantes para o modelo XGBoost de Cooling Load foram: 1. Overall\_Height (0.3289) 2. Relative\_Compactness (0.2765) 3. Surface\_Area (0.1621) 4. Glazing\_Area (0.1234) 5. Wall\_Area (0.0587)

#### 6.2.2 Análise de Resíduos

A análise de resíduos do modelo XGBoost para Cooling Load mostrou resultados similares ao modelo de Heating Load, confirmando a adequação do modelo aos dados.

## 7. Desempenho por Categorias

### 7.1 Desempenho por Orientation

Analisamos o desempenho do modelo para diferentes orientações (Norte, Sul, Leste, Oeste) e identificamos: - Desempenho consistente entre as diferentes orientações - Ligeira melhora no desempenho para a orientação Sul para ambas as variáveis alvo

### 7.2 Desempenho por Glazing Area Distribution

Observamos diferenças mais significativas no desempenho para diferentes distribuições de área envidraçada: - Configurações uniformes tendem a ter previsões mais precisas - Configurações com vidros concentrados em uma única fachada apresentam erros ligeiramente maiores

## 8. Conclusões e Recomendações

### 8.1 Modelos Recomendados

Com base em nossa análise abrangente, recomendamos: - **Para Heating Load**: XGBoost otimizado (R² = 0.9945) - **Para Cooling Load**: XGBoost otimizado (R² = 0.9952)

### 8.2 Justificativa da Escolha

Os modelos XGBoost foram selecionados devido a: 1. **Desempenho superior** em termos de todas as métricas avaliadas (RMSE, MAE, R²) 2. **Tempo de treinamento razoável**, tornando-o prático para implementação 3. **Capacidade de capturar relações não-lineares** nos dados 4. **Robustez** frente à multicolinearidade presente nos dados 5. **Interpretabilidade** através da importância das features

### 8.3 Limitações

Os modelos apresentam algumas limitações: - Desempenho pode variar para edifícios com características muito diferentes das encontradas no conjunto de dados - Dependência crítica de algumas features como Overall\_Height e Relative\_Compactness

### 8.4 Próximos Passos

Para a implementação na aplicação web Django (AT3), recomendamos: 1. Integrar o preprocessador e os modelos XGBoost para ambas variáveis alvo 2. Implementar validações para garantir que os inputs estejam dentro dos intervalos esperados 3. Incluir visualizações