Indicium – **Lighthouse** – Relatório Desafio Ciência de Dados

Desenvolvido por: **Raimundo Renato de Melo Neto**
Link do Repositório: https://github.com/renatom01/predict_wage
Email: rdemeloneto2@gmail.com

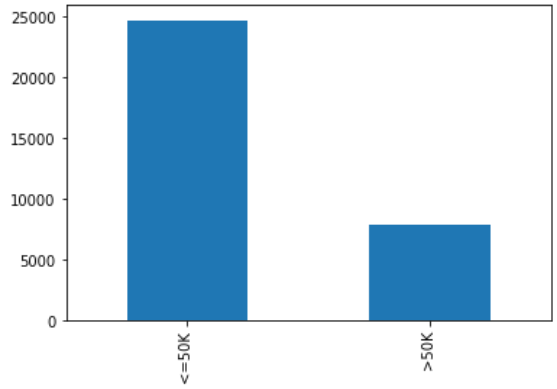
1. Objetivo

Através de modelos Machine Learning, predizer a feature yearly_wage da dataset wage_test.csv utilizando para treinamento os dados contidos em wage_train.csv.

2. Visualização de Dados

A partir de visualização do histograma da feature a ser predita (Figura 1), percebe-se duas classes de salário anual, aqueles que recebem maior valor ou igual a 50.000 (<=50k) e os que recebem um valor menor a este (>50k). Consequentemente, como o objetivo é predizer a que classe uma determinada instância pertence, estamos lidando com um **problema de classificação**.

Figura 1: Histograma da feature yearly_wage.



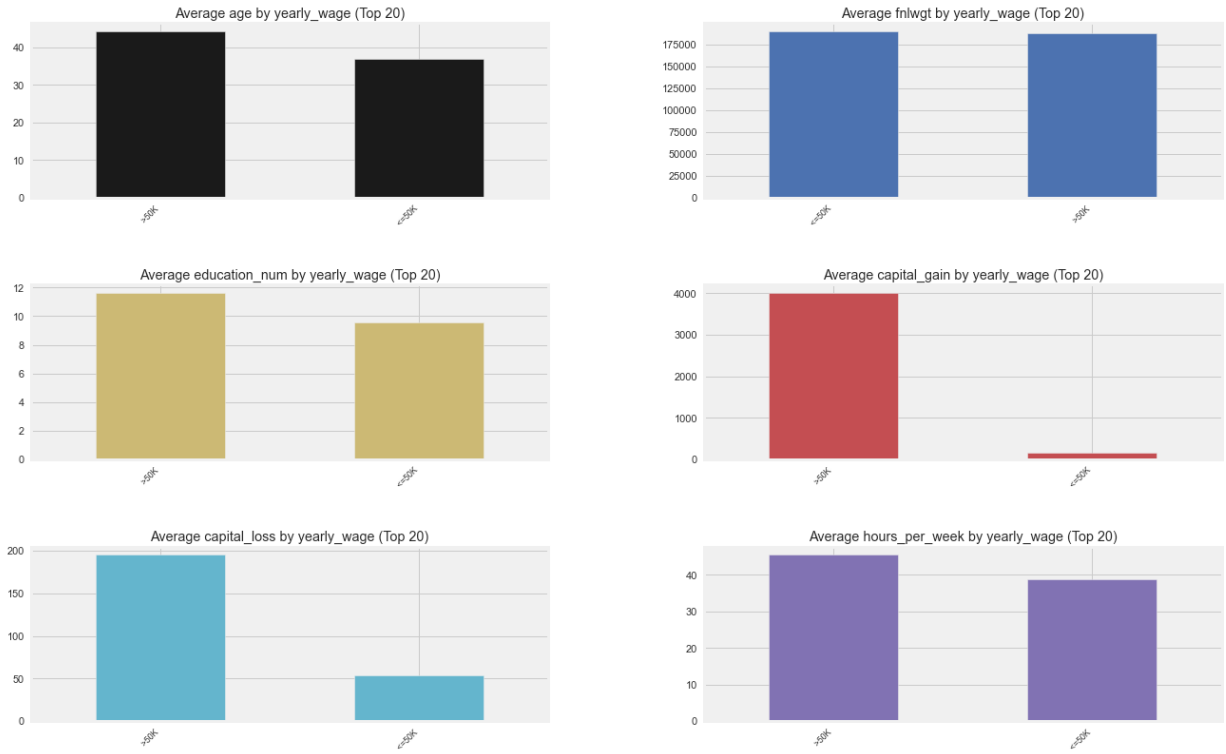
Além disto, a partir a figura 1 é claro o problema de **class imbalance**, muito comum em dados reais, em que uma das classes (<=50K) se sobressai em quantidade de instâncias em relação a classe (>50k), o que posteriormente pode tornar os modelos de Machine Learning enviesados. Porém, no caso deste projeto o imbalance pode ser considerado leve (1:25), não sendo necessário métodos de tratamento como o resampling.

Para melhor compreender a relação entre parte das variáveis independentes e a variável depende (yearly_wage), as features numéricas foram visualizadas em relação a feature yearly_wage. Porém

Na figura 2 é possível obter algumas informações relevantes:

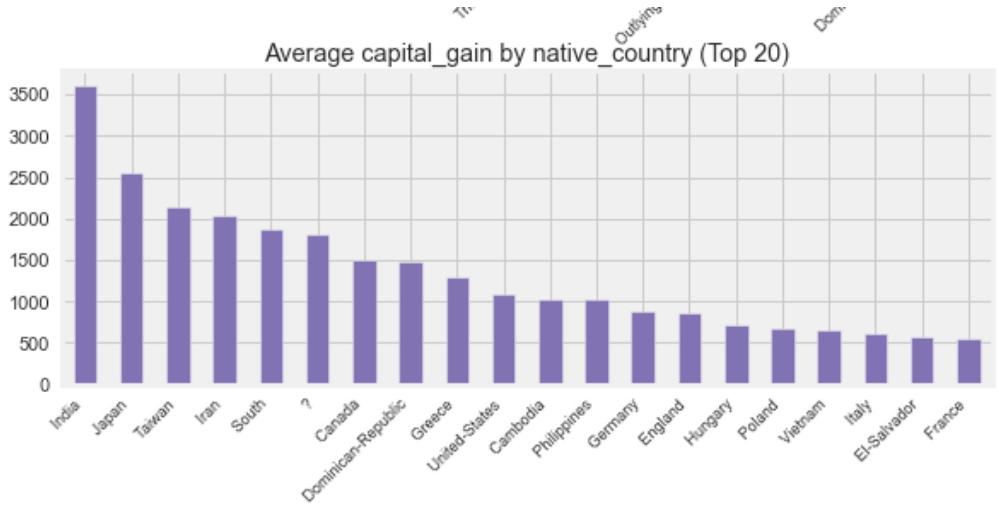
- Aqueles que recebem menos de 50K anualmente, possuem um maior acúmulo de capital, cerca de 40 vezes maior, quando comparados aos que recebem mais de 50K;
- Da mesma forma que acumulam capital, os > 50k (menos de 50k) tendem a perde-lo, porém numa proporção muito menor a que ganharam (em média de 5% do capital acumulado é perdido);
- Nas outras features numéricas, idade, anos de educação e horas de trabalho ambos >50K e <=50K tiveram valores similares, a população consultada tem em média 40 anos, com cerca de 12 anos de educação.

Figure 2: Relação entre yearly_wage e features numéricas.



A partir da figura 3 é possível criar algumas hipóteses das informações extraídas na figura 2, maior acúmulo de capital é percebido na Índia, país considerado de economicamente emergente com oportunidades de crescimento e acúmulo de capital. Seguidos por Japão, Taiwan e Irã, todos no continente asiático, o que explica o fato de salário anual ser menos de 50k dólares, tando devido a oscilação da moeda quanto ao salário anual nestes países.

Figure 3: Acúmulo de capital por país.

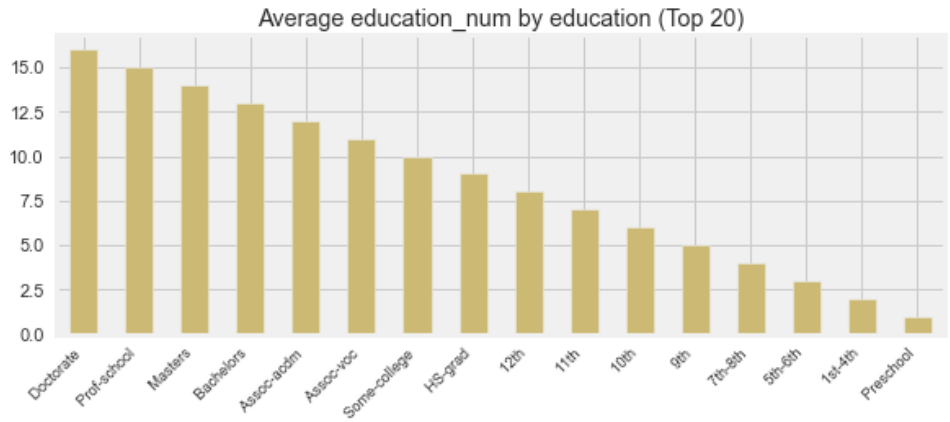


3. Pre-Processamento de Dados

Os seguintes passos foram adotados para preparar os dados: **Data Cleaning, Encoding e Scaling**.

3.1. Data Cleaning

A figura abaixo mostra relação linear entre education e education_num, assim a feature education foi deletada para evitar duplo viés de educação no desenvolvimento do modelo.



3.2. Conversão de Dados (Encoding)

Para converter as features categóricas em numéricas, foi utilizado o algoritmo ordinal encoder.

3.3. Normalização/Scaling de Dados

Modelos de Machine Learning não funcionam de maneira adequada com features que variam muito de escala entre elas, podendo criar um viés àquelas de maior valor. Assim, as features categóricas pós-encoding e a numéricas foram convertidas para uma escala 0-1 através do algoritmo min-max scaler.

4. Desenvolvimento do Modelo

4.1. Data Spliting

O **treinamento** dos modelos neste projeto é do tipo **supervisionado**, assim a feature objetivo (yearly_wage) necessita ser explicitamente identificada, como esta não é presente no dataset wage_test, foi necessário separar os dados presentes em wage_train.csv em 80% para treinamento e 20% para cross-validação dos modelos.

4.2. Cross-Validation

Após treinamento com 80% dos dados presentes em wage_test.csv, os modelos foram submetidos a teste com dados não utilizados no treinamento (20% do wage_test.csv).

4.3. Ajuste de Hiperparâmetros/Regularization

Para evitar o overfitting aos dados de treinamento, os modelos de Machine Learning tiveram seus hiperparâmetros ajustados.

5. Performance dos Modelo

Os modelos de Machine Learning selecionados foram dois modelos tradicionais, Decision Trees, Logistic Regression Classifier acompanhados por ensembles, Random Forest e Adaboost. Diferentemente de regressão, modelos de classificação devem ser avaliados por alguns parâmetros estatísticos, sendo eles:

- **Accuracy:** indica a proximidade do valor predito é do real;
- **Precision:** indica quão próximos ou dispersos estão os dados preditos;
- **Recall:** matematicamente falando, é proporção de verdadeiros positivos identificados corretamente;
- **F1 score:** métrica de combinação entre precision e recall.

Por haver um class imbalance, como demonstrado na introdução, todos os modelos apresentaram um baixo recall da classe minoritária (>50k), assim, avaliar somente a accuracy como performance do modelo é equivocado. Portanto, para este projeto o F1 score melhor se adequa, já que também indica a quão apropriada foi a predição da classe minoritária, sendo o ensemble Adaboost o melhor modelo e posteriormente aplicado para prever a feature yearly_wage do data set wage_test.csv

Modelos ensembles são comumente aplicados por haver alta performance devido a combinação (ensemble) de modelos de machine learning, no Adaboost, decision trees foram treinadas sequencialmente, cada uma tentando corrigir a predição da anterior. Hiperparâmetros como learning rate, max depth foram adequadamente ajustados para obter o máximo F1.

Table 1: Performance dos modelos na cross-validation.

Modelo	Accuracy	Precision		Recall		F1	
		<=50k	>50k	<=50k	>50k	<=50k	>50k
Decision Trees	0.85	0.88	0.75	0.93	0.61	0.91	0.67
Logistic Regression	0.82	0.84	0.70	0.94	0.44	0.88	0.54
Random Forest	0.87	0.89	0.78	0.94	0.64	0.91	0.70
Adaboost	0.87	0.90	0.78	0.94	0.67	**0.92**	**0.72**