Desafio BB - Banco do Brasil

DataSet (Udacity - Ron Kohavi e Barry Becker)

Introdução

O DataSet mostra a renda dos indivíduos usando dados coletados do Censo de 1994 nos EUA. Você deverá escolher o melhor algoritmo candidato, a partir de resultados preliminares, e otimizar mais esse algoritmo para melhor modelar os dados. Seu objetivo com essa implementação é construir um modelo que prevê com precisão se um indivíduo ganha mais de US $ 50.000.

O que deve ser feito

Compreender a renda de um indivíduo pode ajudar uma organização a validar seu poder de compra, verificar se o usuário é um excelente comprador e não é um possível devedor.

Os dados são de pessoas com um valor abaixo de 50k, igual a 50k, ou os dois, ou acima de 50k, sendo essa a nossa variável maior para o processo de classificação. A ideia do desafio é verificar a melhor acuracidade e log loss que o desenvolvedor pode chegar com o DataSet.

O desafio pode ser feito em Scala, Java, R, Python, onde o desenvolvedor deve escolher. Colocar o código fonte e demais artefatos produzidos no desafio em seu repositório pessoal do GitHub.

Deve-se detalhar todos os pontos em um TXT ou Markdown: o que foi usado e o porquê de ter sido usado, quais as variáveis o desenvolvedor levou em conta para a realização do desafio, o que o desenvolvedor fez na realização deste desafio para explorar os dados, transformação dos dados, desempenho e predição, utilização de modelos supervisionados, implementar o treino dos dados para os modelos, modelo de evolução dos dados e a acuracidade.

Detalhes

1. Coluna age: idade da pessoa que participou da avaliação de quanto ganha anualmente;
2. Coluna worclass: é qual o setor que a pessoa trabalha;
3. Coluna education_level: qual é o nível de instrução do usuário, segue abaixo os tipos que serão encontrados e válidos:
    a. Bachelors;
    b. Masters;
    c. HS-grad;
    d. Some-college;
    e. Doctorate.
4. Coluna education_num: número da educação, é o processo de quantificar o nível educacional e dar um peso para seu identificador;
5. Coluna marital-status: o status de matrimônio do usuário que foi entrevistado;
6. Coluna occupation: ocupação do usuário que foi entrevistado, vale lembrar que essa variável pertence ao estado irregular;
7. Coluna relationship: é a coluna de relacionamento, onde detalha seu grau de relacionamento atual, segue as variáveis válidas:
    a. Not-in-family;
    b. Husband;
    c. Wife;
    d. Own-child;

    e. Unmarried.
8. Coluna race: coluna que demonstra a tonalidade da pele do usuário, as variáveis válidas são:
    a. White;
    b. Black;
    c. Asian-Pac-Islander;
    d. Amer-Indian-Eskimo;
    e. Other.
9. Coluna sex: o sexo da pessoa (Masculino ou Feminino);
10. Coluna capital-gain: capital ganho ou adquirido;
11. Coluna capital-loss: capital perdido ou não conquistado;
12. Coluna hours-per-week: horas por semana trabalhadas;
13. Coluna native-country: País nativo do usuário que foi entrevistado;
14. Coluna income: a renda anual do usuário que foi entrevistado, saídas:
    a. Menos que 50k;
    b. Menor e igual a 50k;
    c. Maior que 50k.

Saídas esperadas

Para este desafio esperamos algumas saídas necessárias, conforme abaixo (via TXT ou Markdown):

- Explicar as variáveis que foram exploradas e os motivos;
- Caso tenha sido necessário novas variáveis, explicar o motivo da criação;
- Explicar o processo de preparação;
- Detalhar se houve algum processo que deve ser considerado transformação de recursos contínuos enviesados e o porquê;
- Explicar qual foi o processo utilizado para normalização dos dados;
- Detalhar qual foi o modelo utilizado para desempenho e predição dos dados;
- Se chegou a utilizar Modelos de Aprendizagem Supervisionados, explique quais;
- Detalhar o que foi utilizado no treino e de evolução;
- Apresentar a acuracidade do modelo e a revalidação do modelo com Log Loss.

O arquivo gerado com as saídas solicitadas acima deve ser armazenados em seu repositório pessoal do GitHub.