Relatório Projeto Final- Ciência dos dados



André Tavernaro

Daniel Zelv Freund

Guilherme Batista

Sumário

Introdução\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

Objetivos\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

Ferramentas utilizadas \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_3

Metodologia\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_6

Resultados\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_11

Conclusão e Aplicações\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_12

Referências\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_13

**Introdução**

Com o surgimento da indústria cinematográfica e seu desenvolvimento acelerado ao longo dos anos, diversas plataformas e produtoras de filmes surgiram para revolucionar esse mercado, produzindo filmes em massa para atender a atual população internacionalizada do mundo atual. A ideia do projeto surgiu justamente desse fenômeno de produção em massa de filmes, em que os filmes recebem notas específicas conforme o sucesso que cada um atinge.

**Objetivos**

A fim de desenvolver as habilidades críticas, destreza no manuseamento de altos volumes de informação e trazer uma maior familiaridade do aluno com o processo que um cientista de dados passa em seu cotidiano. O projeto trouxe a oportunidade de se utilizarem regressões, classificadores ou clusters para fazer previsões sobre novas informações.

O tema escolhido foi a regressão linear, usada para prever as notas do IMDB (um dos mais relevantes sites de avaliações cinematográficas mundial). Devido ao uso de diversas variáveis em sua análise, trata-se de um modelo de *Regressão Múltipla*.

**Ferramentas Utilizadas**

* **Regressão Linear Múltipla**

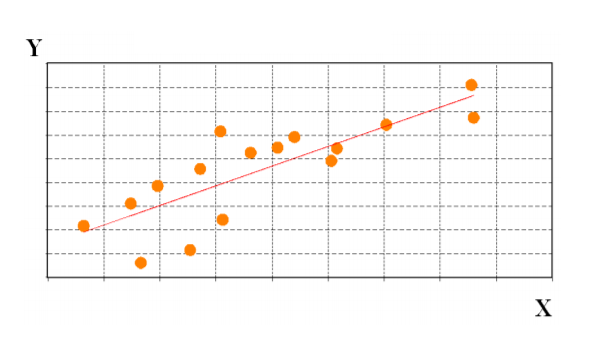
A regressão linear múltipla funciona da seguinte forma: primeiro escolhemos uma variável que queremos prever, depois, coletamos variáveis que possuem forte relação com a escolhida (com comportamento linear), e assim, conseguimos prever de certa forma a variável escolhida para outros dados.

Genericamente, um modelo de regressão linear múltipla com k variáveis independentes e p parâmetros (p=k+1) pode ser representado por:

+ … + +

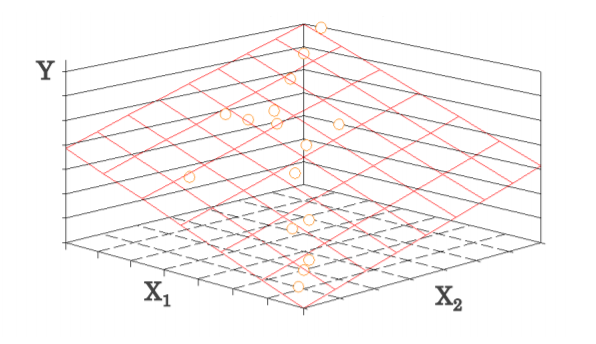
* é o valor esperado de Y quando todos as variáveis independentes forem nulas;
* é a variação esperada de Y dado um incremento unitário em , mantendo constantes todas as demais variáveis independentes;
* é a variação esperada de Y dado um incremento unitário em , mantendo constantes todas as demais variáveis independentes;
* é o erro não explicado pelo modelo;

Gráfico de uma **regressão linear simples:**



Note que o gráfico é **bidimensional**. O modelo tenta ajustar uma reta que melhor se adeque ao comportamento dos pontos, tentando reduzir ao máximo os resíduos.

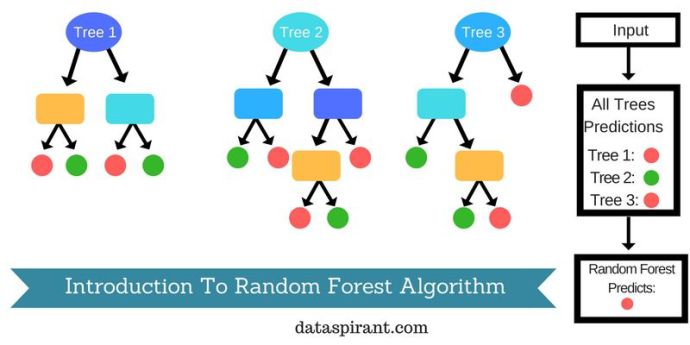
Gráfico de uma **regressão linear múltipla:**



Neste caso o modelo de regressão linear múltipla aborda três variáveis, logo o gráfico possui um plano para cada variável. Tendo sua regressão representada pelo plano entre as três variáveis (X1, X2, Y).

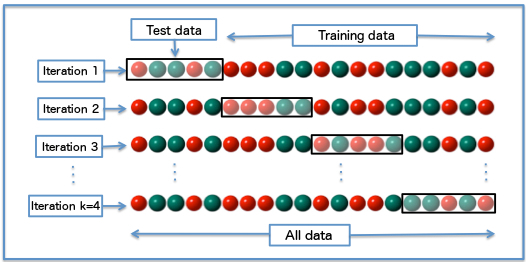
* **Random Forest**

Usado tanto para problemas de regressão como classificadores, o Random Forest cria a partir de um conjunto de dados de treinamento regras de tomada de decisão e verifica se onde os valores de teste mais se adequeam dentro deles.



* **Cross Validation**

Este método divide nosso dataset em várias partições, realizando um teste e um treinamento para cada partição e, obtendo uma precisão para cada uma. Deste modo, fizemos dez partições e calculamos a média das precisões das dez partições, obtendo assim, uma precisão maior para nosso modelo.



* **Bootstrapping**

**Metodologia**

* **Limpeza dos dados**

Para iniciar a limpeza dos dados, primeiro foram removidos os dados duplicados e escolhidas as colunas relevantes para nossa análise. **["title","runtime","budget","revenue","vote\_average","vote\_count","popularity" ,"genres",'release\_date']**

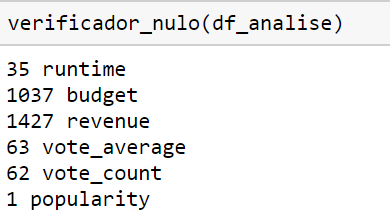
Depois, por se tratar de uma regressão, **genres** e **release\_date** foram convertidos para valores numéricos:

A coluna **genres** contém um dicionário que informa os gêneros em que o filme se encaixa. O problema disso é que os dados não são numéricos. Desse modo foi criado uma coluna para cada gênero, com 0 sendo quando um filme não se encaixa na categoria e 1 quando se encaixa.

A coluna **release\_date** contém a data em que o filme foi lançado, porém ele se encontra no formato “DD-MM-YY, fazendo com que o compilador a interprete como uma string. Logo ela foi particionada em três colunas : Ano, Mês, Dia (integers).

* **Limpeza de valores nulos (0) e inválidos (NA)**

Ao se analisar os valores nulos em cada uma das colunas selecionadas, notou-se uma relevante quantidade de valores nulos:



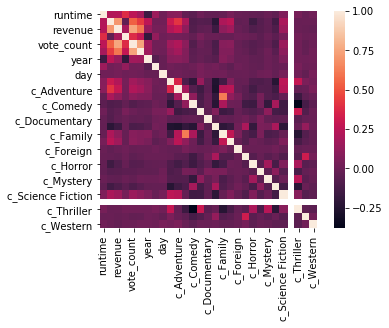
Devido à alta quantidade de valores nulos de*budget*e*revenue*, foi-se optado por **remover** seus filmes do dataframe.

Ao se removerem os valores restantes de *vote\_average*e*vote\_count*, notou-se uma **queda** no de nosso modelo, e ao substituí-los pela mediana de suas colunas, **não** se alterou, portanto elas foram **intocadas** no restante do projeto.

* **Variáveis do dataframe (formatada)**

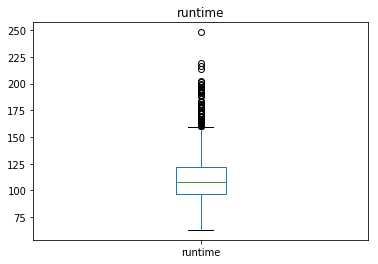
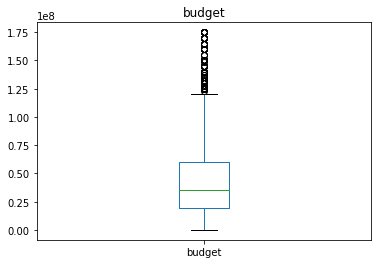
|  |  |
| --- | --- |
| **Title** | **Título do filme** |
| **Runtime** (minutos)  Contagem total: 3164  desvio padrão: 20,67  média: 110,26  mínimo: 41  máximo: 338 | **Tempo de filme** |
| **Budget** (dólares)  Contagem total: 3164  desvio padrão:3.666 e+07  média: 3.706 e+07  mínimo: 3.666 e+07  máximo: 1.750 e+08 | **Orçamento** |
| **Revenue** (dólares)  Contagem total: 3164  média: 1.083 e+08  desvio padrão:1.520 e+08  mínimo: nan  máximo: 1.513 e+08 | **Receita** |
| **Vote\_average** (valor absoluto, média do número de votos)  Contagem total: 3164  média: 6,30  desvio padrão: 0,876  mínimo: 0,00  máximo: 8.5 | **Média dos votos** |
| **Vote Count** (valor absoluto, número de votos)  Contagem total: 3164  média: 903,08  desvio padrão: 1280,86  mínimo: 0,00  máximo: 13752 | **Número de votos das pessoas** |
| **Popularity** (índice que varia de 0 a 100)  Contagem total: 3164  média: 27,84  desvio padrão: 35,10  mínimo: 0,02  máximo: 87,5 | **Popularidade** |
| **Genres** (dicionário com as categorias em que o filme se enquadra) | **Gêneros de filme** |
| **C\_Gênero\_x** ( categorias do filme divididas) | **0 pertence**  **1 não pertence** |
| **Ano** (1916-2016)  Contagem total: 3164  média: 2001,49  desvio padrão: 13,33  mínimo: 1916  máximo: 2016 | **Ano em que o filme foi lançado** |
| **Mês** (1-12)  Contagem total: 3164  média: 7  desvio padrão: 3,37  mínimo: 1  máximo: 12 | **Mês em que o filme foi lançado** |
| **Dia** (1-31)  Contagem total: 3164  média: 15,49  desvio padrão: 8,47  mínimo: 1  máximo: 31 | **Dia em que o filme foi lançado** |

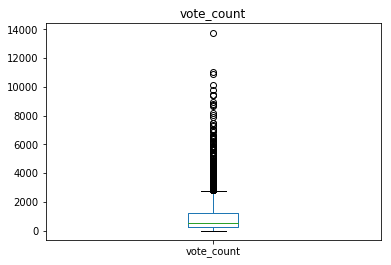
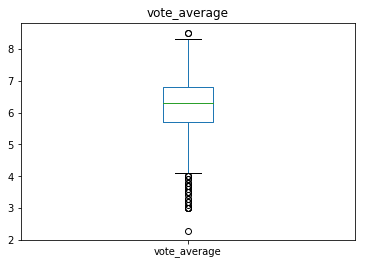
* **Análise gráfica**
* **Heatmap**

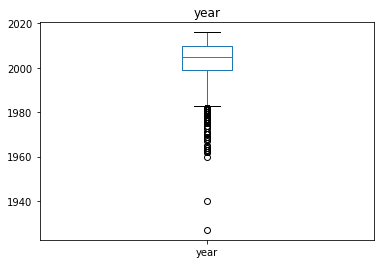
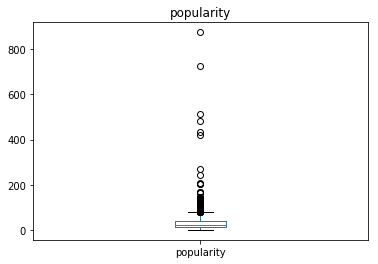
****

Com a ajuda do gráfico *heatmap*, é possível notar o padrão de regressão nas variáveis observadas, permitindo a continuação da análise exploratória.

* **Boxplot**

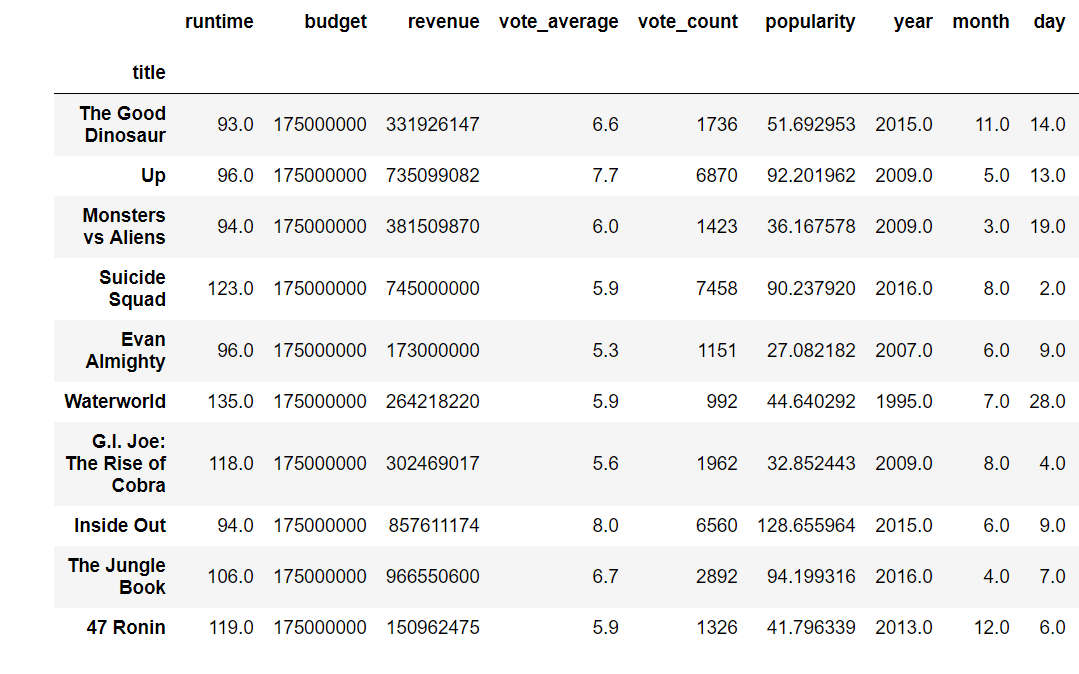
** **

****

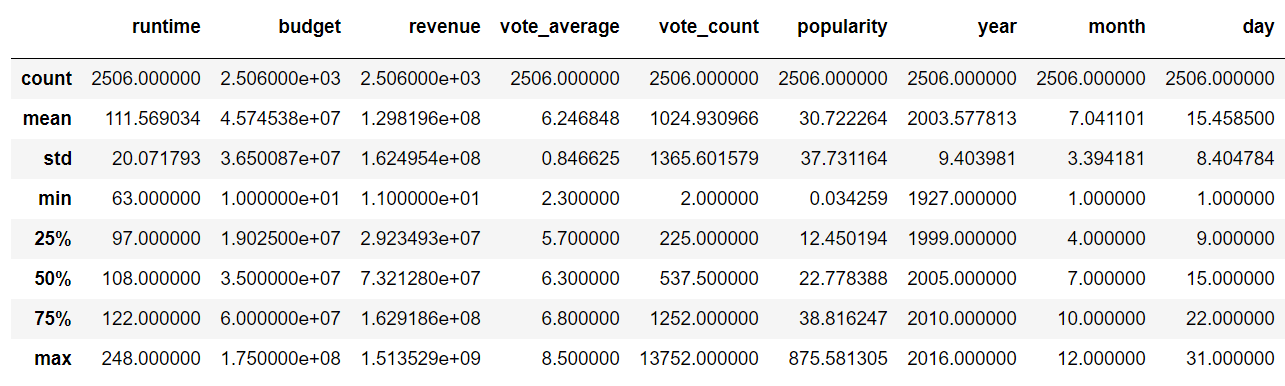
****

Com os **boxplots**, é possível se notar a alta quantidade de outliers em todas as variáveis de nosso modelo.

Em algumas delas, como *revenue* e budget, por se tratarem de variáveis com valores muito altos, foram testados modelos logarítmicos para tentar aumentar a precisão do modelo, mas como isso não ocorreu, foram mantidas **inalteradas**. Ao mesmo tempo, ao se removerem **98%** dos valores mais altos de budget, obteve-se um aumento substancial em **.**



* **Describe**

****

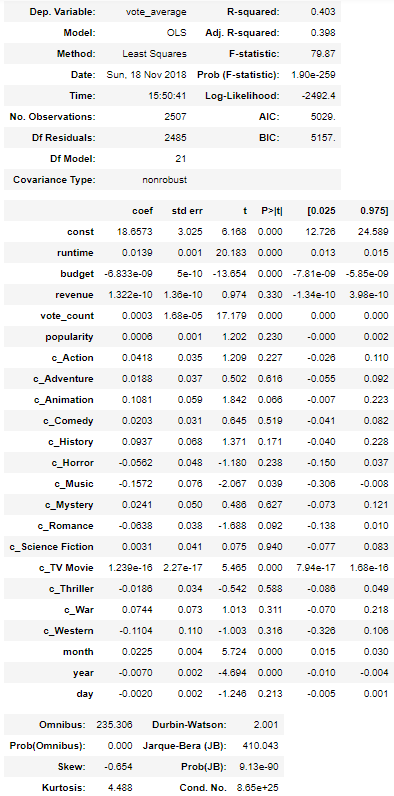
* **Separação das variáveis**

As variáveis de nosso modelo foram separadas em:

1. **Variável dependente (Y)** – Variável da previsão (vote\_average).
2. **Variável independente (X**) – Variáveis relacionadas a varíavel dependente. (nota do imdb)

* **Regress**

Com o uso da função *regress e summary*, e adotando um *alpha* de **5%**, as variáveis que não obtiveram seu P>|t| mínimo foram descartadas. (*c\_genero* e *day*)



* **Divisão TESTE | TREINAMENTO**

Com as variáveis limpas e selecionadas, os filmes foram separados em teste e treinamento (1/3 teste | 2/3 treinamentos). Fazendo com que o algoritmo não se auto alimente, distorcendo os resultados.

**Resultados**

* **Método Regressão Linear: 0.43**
* **Método Random Forest: 0.49**

**Cross Validation ():**

* **Método Regressão Linear:**
* **Método Random Forest: 0.49**

**Bootstrapping():**

* **Método Regressão Linear:**
* **Método Random Forest: 0.49**

**Conclusão e Aplicações**

Depois dos dados serem limpos e analisados, foram aplicados três modelos: *OLS, Random Forest* e *Cross Validation*. Devido ao seu melhor desempenho*, Random Forest* foi o escolhido como primário para nosso algoritmo. Entretanto sua precisão foi de apenas **49%**. Tendo como principal causa a **falta de dados** causados por valores nulos ou inválidos, que acabou limitando a base de dados de **5000 filmes.** Como as variáveis possuíam valores muito diferentes, seriam necessárias quantidades mais altas de dados para análise. Outro fator que deve ser levado em consideração foram as datas dos filmes que se iniciavam desde 1916, que possuíam bilheterias, receitas e orçamentos diferentes dos atuais. Entretanto devido a falta de dados, a remoção destes restringiria muito o campo de análise.

Devido a sua alta aplicabilidade por parte tanto de cineastas para checar sua possível nota em um site de alta credibilidade como o IMDB, como de fãs para ver como seria o próximo filme de sua saga favorita, seria possível fazer uma futura iteração com uma arrecadação de fundos, a fim de obter mais dados para que a análise se tornasse mais precisa. Já que o site disponibiliza um datasheet com mais de 200k de filmes, porém possui um custo muito alto para sua utilização.

**Referências**

<http://www.portalaction.com.br/analise-de-regressao/regressao-linear-multipla>

https://www.surveyanalytics.com/system/heatmap.html

https://www.kaggle.com/adktyakirloskar/movies

http://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/

https://scikit-learn.org/stable/modules/cross\_validation.html