Data science case study: Analyzing the script of the Office

Introduction

The office is one of the most successful sitcoms in TV history, and has become a widely acclaimed cult classic. Boasting an overall rating of 9.0/10.0 on IMDB it is clear that the general sentiment towards the office is very positive. However, not every episode is such a fan favorite, while some episodes achieved an IMDB rating of an almost perfect 9.8/10.0, others scored as low as a 6.4/10.0. The goal of this case study is to use the script of the office to pick up indicators of whether an episode will achieve a high rating. These indicators will provide insight into what makes the office tick. What characters are fan favorites? What characters are not so liked? What seasons really stood out in the ratings? Does the overall sentiment of the script influence how the episode is perceived by the audience? These are all examples of some questions that you should aim to answer by processing the Schrute Dataset.

The schrute dataset is a collection of the scripts of every single office episode, including the lines spoken by each character, as well as additional metadata about the episode such as IMDB rating, director, and air date. Each of these columns can be used to perform an interesting analysis on their own, but combined, can create even more creative insights. Examples of analysis could be how the number of lines a character has changed over time, if certain directors/writers tend to use different characters more or less often, or how the sentiment of a character's lines impact the rating of the episode.

After understanding what factors have influenced the shows direction and ratings over time, you can then create a regression model that is able to use various features of the dataset to predict the rating of the show. This insight could then be used to gain a comprehensive picture of what drove the office to its huge success, as well as what the writers could have done to improve the ratings on the less successful episodes

Overall, the deliverable for this project should summarize the results of the analysis, and provide a statement on what worked, and what did not work in terms of the offices' writing. Being such an influential and popular show, this should be a fun project to explore the structure of sitcoms.

Rubric

Formatting	● Written portion
	  o Submit the written portion as a PDF
	  o The submitted document should contain a consistent font and include headings for each section
	  o Proper mechanical and grammar rules should be used in writing
	● Data and Code
	  o All data and code used for the analysis should be submitted in a github repository
	  o The github repository should have a readme that contains an overview of the project, and steps for reproduction.
	  o The repository should have the following folders and files
	  ▪ DATA
	  ▪ SCRIPTS
	  ▪ OUTPUT
	  ▪ License
	  ▪ Readme.md
	● References
	  o References should be included in the written portion of the report
	  o All references should be cited in IEEE format

Written Portion	● Goal: This pdf should make a 2nd year UVA student excited to engage with your case study. ● Create a scenario and place the student in the 'driver's seat' ● Make clear the topic/context/motivation ● Indicate what deliverable the student is to produce but withhold details, those come in the rubric. This is a high level mission document not a specifications document. ● One page maximum ● PDF format
Data	● All data used in the project should be stored in the DATA directory of the github repository o If the data is not allowed to be redistributed (for licensing or other reasons) instructions on how to retrieve the data should be included in the DATA directory ● A data appendix should be included that lists all columns in the dataset, as well as a brief explanation and expected values
Code	● All code needed to reproduce the analysis should be stored in the SCRIPTS folder

	● Instructions on how to install any software needed to perform the analysis should be included in a setup.md file in the SCRIPTS directory
	  o This should also include a list of libraries/dependencies. If using a package management system (pip, npm, etc) a requirements file (requirements.txt, package.json) should also be included as well as the command needed to install all dependencies
	  o If there are any approach agnostic decisions (such as editor used to open the code) a recommended approach should be provided, as well as a note that it can be performed with other software.
	● The code should take care of any data cleaning and exploratory analysis.
	  o All data cleaning decisions should be documented either as comments or as a markdown header (jupyter notebook, Rmd file)
	  o Exploratory data analysis should contain information about the dataset (columns & rows) as well as a minimum of two charts that give an overview of the dataset, and motivate further exploration
	● The analysis should include at least one regression analysis that uses features from the dataset to predict the ratings of an episode
	  o If possible, explanations of the regression should be included (weights of a linear regression, or influential nodes of a decision tree)

Readme and Documentation	● The README.md file should contain
	  ○ An overview of the repository contents
	  ○ Motivation for the project
	  ○ A summary of the analysis
	  ○ Steps for reproducing the results
	● All citations in the written portion should be in IEEE format
	● Documentation should be clear to understand for the intended audience
	● A license should be included that is no more permissive than any of the software or data used in the analysis.

Github Repository

Link: https://github.com/GarrettBurroughs/DS4002-CS3

Here is a github repository containing the data to get you started, as well as some starter code. The readme will be a useful place to start. While the dataset is given by an R package (and a .rda file) code has also been included to import the file into python using a pandas dataframe. The repository also contains some more holistic analysis of the office and the characters within it. These blog posts may inspire some analysis approaches.

Repository outline

DS4002-CS3
├── **DATA**
│ └── **theoffice.rda** - The dataset in .rda format
├── **LICENSE** - MIT license
├── **materials**
│ ├── **blog_posts** - Blog post analyses of the office
│ │ ├── **An Exploratory Character Analysis of Michael Scott by Jordan Sprogis Medium.pdf**
│ │ └── **The Office A Final Critical Analysis – Alyssa's Blog.pdf**
│ └── **reference_scripts** - Scripts for loading in the data in Python and R
│ ├── **loadPython.ipynb**
│ ├── **loadR.html**
│ └── **loadR.Rmd**
├── **Project Outline and Rubric.pdf** - The project motivation, outline, deliverable, and rubric
└── **README.md** - Overview of the repository and Case Study

References

Here are some references that you may find helpful in this case study.

- The Schrute dataset: https://github.com/bradlindblad/schrute
- The VADER sentiment analysis package: https://github.com/cjhutto/vaderSentiment
- Pandas - A python data analysis package: https://pandas.pydata.org/
- Scikit Learn - A python machine learning package: https://scikit-learn.org/stable/