Take-home Technical Assessment

Bayesian Statistical Modeling of the Twitter Airline Sentiment Dataset

Rocco Vivier
basileusza@gmail.com
Stellenbosch, Western Cape, South Africa
GitHub: https://github.com/yesteryearer/Praelexis-Technical-Assessment
LinkedIn

Abstract

This technical report present a novel exploration of the Twitter Airline Sentiment dataset, leveraging the power of probabilistic programming with PyMC, a Python library for Bayesian statistical modeling. The primary objective of the study is to create a minimal model of selected aspects of the data, demonstrating an understanding of the capabilities of PyMC in data science applications. Two models were created on the dataset, a naive Bayes model based on the airline sentiment and a multivariate regression model also based on the airline sentiment. The report delineates the research results uncovered in employing the tool, as well as the general thought process behind the work. Emphasis is placed on the rationale behind choosing specific modeling approaches, the selection and interpretation of metrics for model evaluation, and critical analysis of the outcomes. The report serves as a testament to the adaptability and problem-solving skills required in data science roles, showcasing the ability to harness new tools effectively and efficiently. The report aims to provide a comprehensive and insightful analysis of the modeling of the dataset, offering a unique perspective on the application of PyMC in data science.

Keywords:

Data Analytics, Probalistic Programming, Bayesian Inference, Bayesian Modeling, PyMC

Contents

1 Background

1.1 Bayesian Inference

Bayesian inference is a method of statistical inference in which Bayes' theorem is used to update the probability for a hypothesis as more evidence or information becomes available. It is an approach to statistical modeling that interprets probability as a measure of believability or confidence that an individual might possess about the occurrence of a particular event. Mathematically, Bayes' theorem is expressed as:

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)}$$

where $P(H|E)$ is the probability of hypothesis H given the evidence E, $P(E|H)$ is the probability of evidence E given that hypothesis H is true, $P(H)$ is the prior probability of hypothesis H, and $P(E)$ is the probability of evidence E.

Baye's theorem is a simple consequence of the definition of conditional probability. It has this name because it is extremely useful for making inferences about phenomena that cannot be observed directly. Sometimes these inferences are described as "reasoning about causes when we observe effects."

1.2 Bayesian Modeling

Bayesian modeling involves constructing statistical models where the inference is based upon Bayesian principles. This approach allows for the integration of prior knowledge along with the observed data to make inferences. Unlike frequentist statistics, which interpret probability solely as the frequency of occurrence of particular outcomes, Bayesian models provide a flexible framework for modeling complex phenomena and making probabilistic statements about unknown parameters.

- **Prior Distribution (Prior)**: The prior distribution, commonly referred to simply as the "prior," represents what is known about the parameters before observing the data. It embodies any existing knowledge or assumptions about the system being modeled, whether from past data, theoretical understanding, or expert opinion.
- **Likelihood**: The likelihood is a function that measures the compatibility of the observed data with different values of the model parameters. In Bayesian inference, it quantifies how likely the observed data is, given specific values of the parameters.
- **Posterior Distribution (Posterior)**: The posterior distribution combines the information from the prior distribution and the likelihood. It reflects the updated beliefs about the model's parameters after considering the observed data. Mathematically, it is derived from the application of Bayes' theorem, as it incorporates both the prior information and the likelihood of the observed data.

1.3 PyMC

PyMC is an open-source probabilistic programming framework written in Python. It is used for Bayesian statistical modeling and probabilistic machine learning. PyMC offers powerful and flexible modeling capabilities, enabling users to construct complex Bayesian models easily. The library utilizes advanced sampling algorithms, such as Markov Chain Monte Carlo (MCMC), to estimate the posterior distribution of the model parameters. PyMC is particularly designed to accommodate the creation of complex models and to work seamlessly with the broader scientific Python ecosystem, including libraries such as NumPy, SciPy, and pandas.

2 Methodology

2.1 Domain Research

Embarking on this endeavor required a foundational understanding of probabilistic programming and Bayesian modeling, areas previously unexplored by me. To bridge this gap in knowledge, an intensive two-day research period was undertaken. This involved a dive into PyMC's comprehensive documentation, augmented by engaging with educational videos on Bayesian inference. Additionally, a thorough review of statistical methodologies was conducted, utilizing my personal collection of statistical textbooks, thus ensuring a well-rounded approach to the forthcoming challenges.

2.2 Exploratory Data Analysis

The initial phase of the analysis focused on gaining a nuanced understanding of the dataset's characteristics. This exploratory data analysis entailed examining various attributes of the dataset, such as the distribution of values, identification of unique elements, and assessment of the proportion of missing data. These insights not only provided a holistic view of the dataset but also acted as a catalyst for the formulation of potential modeling strategies.

2.3 Data Pre-processing

The data pre-processing stage addressed several critical aspects to refine the dataset for effective modeling. This included meticulously handling missing values, employing category encoding for classification features, and standardizing the data for uniformity. A novel approach was also adopted by transforming the timestamp data from tweet creation into fractional hours, subsequently converting this temporal information into sine and cosine representations. This transformation was aimed at capturing the cyclical nature of time-related features in the data, thus enriching the model's contextual understanding. The

initial pre-processing was geared at remaining as general and interpretable as possible. Further pre-processing was conducted on a model-specific basis.

2.4 Modeling

The modeling phase was characterized by an exploratory approach, where a spectrum of models, varying in complexity, was examined [1]. This strategy was not only instrumental in familiarizing myself with the nuances of the PyMC package but also in appreciating the intricacies of Bayesian modeling. The objective was to not only develop a functional model but to also gain an understanding of the underlying statistical processes.

3 Model Descriptions

3.1 Naive Bayes Model

The first model is a Naive Bayes model, a probabilistic classifier that applies Bayes' theorem with strong independence assumptions between the features. In this model, the key components are:

- **Class Prior:** A Dirichlet distribution is used to estimate the prior probabilities of the classes, reflecting our initial beliefs before observing the data. In this case, the sentiments associated with the airline.
- **Feature Distributions:** The features' distributions are modeled as Normal distributions for their means and Half-Normal distributions for their standard deviations. This approach captures the central tendency and variability in the features for each class.
- **Feature Likelihoods:** The likelihood of observing the features given a particular class is modeled using a Normal distribution. This step is crucial in updating our beliefs based on the observed data.

This model is particularly adept at capturing the underlying relationships between the independent features and the target variable, assuming that each feature contributes independently to the probability of the target.

3.2 Multivariate Regression Model

The second model is a multivariate regression model, designed to capture complex relationships between multiple predictors and the response variable. It includes:

- **Linear Predictors:** The model constructs a linear relationship involving various predictors, including categorical variables (like airline), and continuous variables (like sentiment confidence and time).
- **Cyclical Nature of Time:** It uniquely incorporates time by transforming it into sine and cosine components, acknowledging the cyclic nature of time-related variables.
- **Parameter Distributions:** The coefficients for each predictor, as well as the intercept, are modeled using Normal distributions. This choice reflects the belief that the true values of these parameters are centered around a mean with a certain variability.
- **Observation Model:** The final airline sentiment observations are modeled with a Normal distribution, encompassing the predicted values and their variability.

The key components of the multivariate regression model are:

- `alpha`: The intercept represents the baseline level of the response variable when all predictors are at their reference levels.
- `airline_coeff`: A normal prior for the coefficients associated with the unique airlines in the data. The model allows for different baseline sentiment levels for each airline.
- `conf_coeff`: A normal prior for the coefficient associated with the sentiment confidence. This term adjusts the baseline sentiment level based on the confidence of the sentiment.
- `sine_coeff` and `cosine_coeff`: Normal priors for the coefficients associated with the sine and cosine transformations of the hour, likely to capture the cyclic nature of time within a day.
- `sigma`: A half-normal prior for the standard deviation of the outcome, reflecting the expected variability in the response variable around the mean.
- `mu`: The deterministic part of the model, combining the intercept, airline-specific adjustments, and the effects of sentiment confidence and time of day on the baseline sentiment level. There might have been sense in using the `deterministic` keyword here I encountered in the documentation.
- `sentiment_obs`: The likelihood function, assuming that the observed sentiment scores are normally distributed around the mean (mu) with a standard deviation (sigma).

This model is adept (or could be, at least, if properly parameterized and fitted) at elucidating the multifaceted relationships within the data, capturing both linear dependencies and cyclical patterns, making it suitable for complex, real-world datasets.

1. These models are contained in the `/models/incomplete` directory of the github repository

3.3 Model Evaluation

Both models are evaluated using advanced sampling techniques like the No-U-Turn Sampler (NUTS), which facilitates efficient exploration of the parameter space.[2] The evaluation focuses on assessing the posterior distributions of the model parameters, providing insights into their confidence intervals and central tendencies. The following metrics are used to analyse the trace produced by the models:

- **mean:** The average value of the posterior distribution.
- **sd:** The standard deviation of the posterior distribution, indicating the spread.
- **hdi_3% and hdi_97%:** The bounds of the 9Highest Density Interval (HDI), giving a range where a percentage of the posterior distribution lies.
- **mcse_mean and mcse_sd:** The Monte Carlo Standard Error for the mean and standard deviation, respectively, which gives an indication of the precision of the MCMC estimates.
- **ess_bulk and ess_tail:** The effective sample size for the bulk of the distribution and the tail, respectively, which measures the number of independent samples that are effectively equivalent to the correlated samples in the MCMC chain.[3]
- **r_hat:** The Gelman-Rubin diagnostic, where a value close to 1 indicates that the chains have converged.

4 Research Results

4.1 Naive Bayes Model

The posterior distributions of the means for each class and feature represents the most likely values for the means, and these appear to be well estimated with some variation between classes which can help in classification tasks. Looking at the R-hat values, all parameters have converged well (values are 1.0), which indicates that the chains have likely converged to the posterior distribution. The convergence of the MCMC chains for the class priors appear to have mixed well, as seen on the plots in the `.ipynb`. The chains are indistinguishable from each other since they are overlapping and horizontal. However, despite this, the wide distributions of the class priors could be an indication that the class priors are not defined well. This suggests that the data does not provide strong information about the relative sizes of the classes.[4] This was also my suspicion during EDA, a better approach would have been to conduct feature engineering based on the actual contents of the tweets.

The effective sample size (ess) for all parameters is high, which is good. It means that the posterior estimates should be quite reliable. The means and standard deviations (sds) for each class and feature have small standard deviations in the posterior estimates, suggesting that the data provides clear information about their values, albeit non-informative. A clear channel doesn't mean the signal is meaningful I guess.

In conclusion, the model appears to be well-specified and the MCMC simulation has converged effectively, given the high effective sample sizes and R-hat values close to 1. The class priors are not well-defined as their distributions are quite wide, suggesting that the data does not provide strong information about the relative sizes of the classes. The means are well estimated, with some variation between classes, which should aid in classification tasks. The standard deviations vary between features but are generally well estimated with narrow credible intervals.

4.2 Multivariate Regression Model

My initial instinct of this model tells me that it suffers from bad parameterization. The distribution of the posterior intercept is wide, an indication of substantial uncertainty associated with the overall baseline sentiment. There is a similar case present with the coefficients associated with each airline. This could be an indication of a great amount of uncertainty and/or just very little information provided by the airlines. This could also potentially be as a result of the way categorical encoding was employed.

Despite best efforts to remedy them, the warnings about the chains reaching the maximum tree depth persisted. They could suggest that the model may be complex or that there are strong posterior correlations, which could make sampling less efficient. Interestingly, this didn't seem to affect the `r_hat` values, which were all 1, meaning the model's chains still converged seemingly adequately. For interesting discussion, see the links in the footnote.[5] The posterior distribution for the sentiment confidence coefficient is narrow and negative, suggesting a consistent and

2. Monte Carlo methods are a class of computational algorithms that rely on repeated random sampling to obtain numerical results. They are particularly useful in situations where solving a problem analytically or through deterministic means is difficult or impossible. This is often the case with complex integrals, especially those that cannot be solved with traditional analytical methods
3. The effective sample size (ESS) is a measure of the number of independent samples that are equivalent to the correlated samples obtained from the Markov Chain Monte Carlo (MCMC) process; a high ESS indicates that the sampling process has produced a diverse and representative set of points from the posterior distribution, reducing the Monte Carlo error and thus leading to more precise and reliable estimates of the parameters.
4. When the data does not provide strong information about the relative sizes of the classes, it's typically because the observed data does not distinctly favor certain class proportions over others, or there is a high degree of overlap between the classes in the feature space.
5. `https://arxiv.org/abs/1903.08008`, `https://discourse.pymc.io/t/what-does-the-maximum-tree-depth-warning-mean/4173/2`

precise negative effect of sentiment confidence on the response variable. During EDA, it was established that most airline sentiments were negative and airline confidence was predominantly at 100%, with some confidence levels centered around the 50% range, therefore, this might be reverse correlation where a skewed distribution of negative sentiments was the primary cause.
The coefficients for the time of day have narrow posterior distributions, suggesting precise estimates, but the effect sizes are small. This result makes sense, since the tweets were pretty evenly dispersed across the day, only diminishing in frequency in the early morning hours.
The posterior for sigma is narrow, indicating that the model is fairly certain about the level of noise in the data. Again, due to the selection of features for the prior distribution that contained, in my estimation, poor informational capabilities, this lack of noise could just be due to the lack of underlying complexity or underlying patterns typical of fine-grained real-world data. This most certainly would not be the case if the a comprehensive feature engineering was conducted on the content of the tweets and used for modeling.

The effective sample sizes (`ess_bulk` and `ess_tail`) are generally high, indicating reliable estimates. However, for the intercept and airline coefficients, they are lower (around 1200), suggesting less certainty or potential sampling issues for these parameters. Similarly, the high density intervals were generally narrow, suggesting precise estimates, except for the intercept and airline coefficients, once again reflecting greater uncertainty. In conclusion, the model captured certain aspects with high precision, but there was a lot of uncertainty in the baseline level, as show by the intercept.

4.3 Reflections on the Assessment

Upon reflection, the selected features for my model appeared to lack substantive descriptive power. The primary distinction among the database records was the content of the tweets, which I neglected to incorporate effectively into my model. In future attempts, I would prioritize the analysis of tweet content, potentially employing multi-dimensional vectors or utilizing natural language processing tools like BERT to generate tokens.

Additionally, my attempt to utilize PyMC's categorical feature for modeling observed airline sentiment was thwarted by challenges in applying the necessary softmax transformation to the prior coefficients. My limited domain knowledge at the time impeded my ability to rectify this issue.

This assessment has led me to appreciate the complexity and depth of Bayesian modeling and inference, which demand both substantial domain expertise and the development of a critical skill set. Throughout the process, I sensed an underlying layer of complexity yet to be explored. The experience has sparked a curiosity in Bayesian methods that I am eager to pursue further in my own time. Bayes' theorem is one of the profoundest afterall.