CORNELL TECH

Poker Bluff Detection using Real Life Court Trial Deception Dataset

Applied Machine Learning Final Project

Rohan Narayan, Maanas Peri, Arystan Tatishev
Cornell Tech, Cornell University, 2 W Loop Rd, New York, NY 10044, USA

Submitted 12 December 2023

Abstract

This study explores the application of machine learning techniques in the detection of bluffing through facial expressions in poker games. Utilizing a dataset of annotated poker game footage, we rigorously tested several machine learning models, including Logistic Regression, Bernoulli Naive Bayes, Linear Discriminant Analysis, SVMs, and Neural Networks, to evaluate their efficacy in identifying deceptive behaviors. The central metric for assessment was the accuracy of each model in distinguishing bluffs from genuine expressions. The Naive Bayes model demonstrated notable proficiency, achieving the highest accuracy at 66.67%, indicating its adeptness in handling the complexities of facial expression classification in bluff detection. This research contributes to the existing literature on deception detection by not only providing a practical application in a challenging real-world context but also by highlighting the potential of relatively simpler models like Naive Bayes in specific analytical tasks. The findings of this study have broader implications, suggesting future research directions in enhancing deception detection methodologies across various high-stakes environments.

Introduction

Motivation

We would like to build a bluff detection system to be used in contexts such as poker or other card games, where a model analyzes a player's facial expressions and determines if they are bluffing or not. This is an application of computer vision classification techniques and builds on existing studies done on the detection of deception in facial recognition. Some studies include "Detecting deception through facial expressions in a dataset of videotaped interviews: A comparison between human judges and machine learning models"[2], "Deception Detection using Real-life Trial Data"[3], and many more cited below. From unique data transforming methods to implementing different computer vision classification techniques, we plan on building upon these studies and seeing if we can achieve similar results.

Background

Prior work in the field of deception detection has primarily focused on facial recognition and interpreting microexpressions. Studies such as "Detecting deception through facial expressions" and "Deception Detection using Real-life Trial Data" have laid the foundation for this project. The project builds on these studies, aiming to translate the findings into practical applications in poker and possibly other scenarios like courtroom settings.

Method

Dataset

The "Deception Detection using Real-life Trial Data" dataset had 39 different gestures that were tracked during court trials, predominantly focusing on facial expressions and body language, as well as the outcome if the subject of the trial was deceptive or not. With this data, we ran three classification models, with the results below:

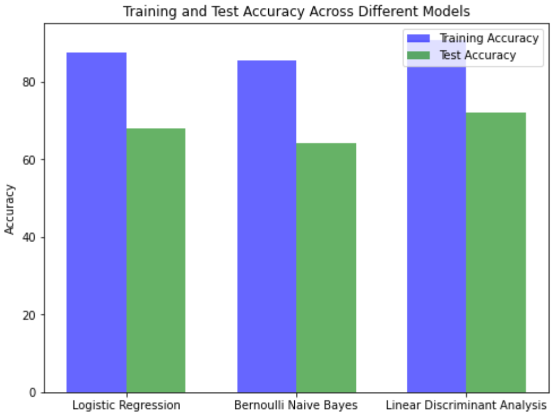


Figure 1: Training and Test Accuracy for Logistic Regression, Bernoulli Naive Bayes and Linear Discriminant Analysis.

We can observe that Linear Discriminant Analysis slightly outperformed the other two models, but all models performed fairly well, especially given the small size of the data source. For Linear Discriminant Analysis, the training accuracy was above 90% while the validation accuracy was above 70%, showing signs of some overfitting. We next decided to analyze the features that were most prevalent for each model, as seen below:

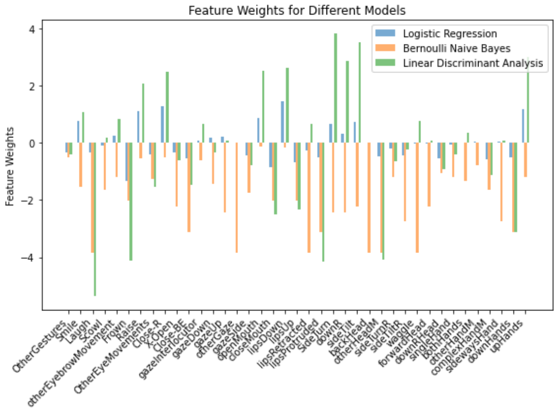


Figure 1: Feature Weights of Courtroom Data for different models

Based on the observations, we were able to determine that some features were not as useful in determining if the court footage was deception or not. For example, “gazeDown”, meaning looking down, was not a strong indicator for any of the models. However, all the models significantly used “Raise”, meaning raised eyebrows, as an indication of deception.

Feature Selection

Based on the weights of the features of the training set, we have opted to only track certain features when collecting data. Some features like Smile, Laughter, Scowl, Frown brow etc., had a bigger impact on the prediction accuracy of the training dataset. Furthermore, during a game of poker, some features are less pronounced than others, which narrowed the feature list to 25 features.

Data Collection

We collected testing data by manually annotating two poker videos where players were evidently bluffing. As an audience, we see the hands that the players have, making it easier for us to annotate if the player was bluffing or not. By separating the selected features between our team members, each member was tracking all the facial expressions and body language of all the active players in the poker round. The annotations were made only when it was the active player’s turn to bet. To make sure that the data is diverse enough, we have sampled the data from both professional poker tournaments[5] and celebrity invitational tournaments[6].

Pre-processing

Since we have opted to include only certain features, our main pre-processing step was just dropping the omitted features from our training dataset. We preprocessed our testing dataset manually by recording facial expressions and cues from poker footage.

Experimental Analysis

Experiment 1

Experiment #1 focused on evaluating machine learning models for deception detection in high-stakes scenarios, using algorithms like Logistic Regression, Bernoulli Naive Bayes, and Linear Discriminant Analysis on a dataset of court case bluffs. Validation accuracies varied, revealing insights into how each model's strengths aligned with the unique characteristics of our dataset.

Overall Accuracy for Logistic Regression: 45.83%
Overall Accuracy for Naive Bayes: 70.83%
Overall Accuracy for LDA: 64.17%
Overall Accuracy for SVM: 66.67%
Overall Accuracy for Neural Networks: 62.50%
Error Analysis?

Bluff	Log. Reg.	NB	LDA	SVM	NNs
Precision	0.37	0.60	0.36	0.50	0.43
Recall	0.88	0.38	0.50	0.88	0.38
F1	0.52	0.46	0.42	0.64	0.40

Truth	Log. Reg.	NB	LDA	SVM	NNs
Precision	0.80	0.74	0.69	0.90	0.71
Recall	0.25	0.88	0.56	0.56	0.75
F1	0.38	0.80	0.62	0.69	0.73

Overall, the error analysis shows higher recall and precision for the truth values, indicating that the model was better at predicting when the subject was telling the truth. However, there were low F1 scores for the bluff class, indicating poor precision and recall. The validation accuracies and error analysis revealed distinctive strengths and weaknesses for each model:

Logistic Regression exhibited a relatively low overall accuracy of 45.83%. This could be attributed to its linear nature, which may not effectively capture the intricate relationships within the facial expressions associated with deception in high-stakes scenarios.

The **Naive Bayes** model performed much better comparatively with an overall accuracy of 70.83%. This ML model is suitable for datasets with binary or categorical features. This fits our criteria since these features are largely independent, meaning that regardless of how these features perform they're not indicative of the other. This property perfectly fits NB assumption of independence. However, this may oversimplify our dataset and create relationships/patterns that aren't necessarily true.

LDA is a statistical method commonly used for classification and assumes that the data distribution of each feature in our dataset is distributed normally. In our case, our categorical features don't follow this distribution. This may be why this model performed 54.17%, which is relatively low compared to our initial Naive Bayes benchmark.

With our accuracies falling short of expectations, we explored more complex models like Support Vector Machines and Neural Networks. The intricate and non-linear nature of facial expressions in deception led us to believe these models might better capture nuanced relationships in the data. Our performance analysis is as follows:

SVM, a classifier akin to logistic regression, effectively separates data using a hyperplane and handles outliers well. It excels in managing non-linear patterns using kernel functions, avoiding the need for feature engineering. This efficiency was evident in our 66.67% accuracy, showcasing its ability to predict complex non-linear relationships. However, SVMs may oversimplify such relationships in smaller datasets, a limitation that could become more apparent with larger datasets.

NN, a complex model, aimed to uncover detailed patterns in our dataset. Utilizing a multi-layer perceptron with a sigmoid activation function, it achieved a notable 62.65% accuracy. However, this fell below expectations compared to other complex models like SVMs, likely due to the small size of our training dataset (123 entries). Neural Networks generally excel in larger datasets, where their multi-layer architecture and nonlinear functions can effectively capture more intricate patterns without introducing excessive complexity for simpler patterns.

Experiment 2

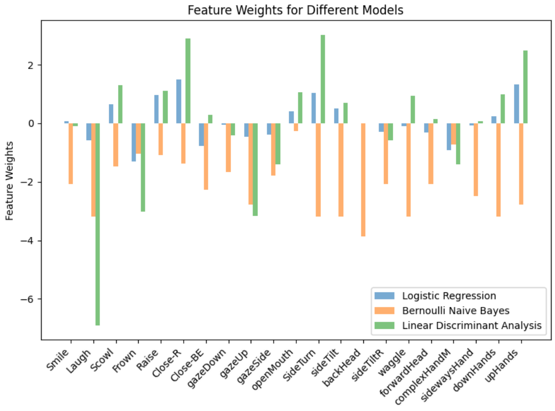
This experiment is about comparing feature weights before and after we dropped features to understand how big/small of weight these models will assign to certain features. Here's a comparison of our top 5 feature weights for each order (in no particular order). Here, the green-colored weights are the top 5 weights that the regression model has kept consistently through and through, suggesting their relevance in capturing deceptive behavior, which is probably an indicator as to why it provided lower ranked accuracies. However, Linear Discriminant Analysis assigned higher weights to a different set of categorical features completely, which could be another reason why it displayed lower accuracies.

Before:	**After:**
Logistic Regression Top 5 Important Feature Weights: Close-R: 0.9777134429584029 Frown: -1.654068332219512 Raise: 1.299502331138967 upHands: 1.1312730535976887 lipsDown: 1.4660941854001663	**Logistic Regression Top 5 Important Feature Weights:** Close-R: 1.483736427797438 Frown: -1.2946596111491684 Raise: 0.961052038902934 upHands: 1.33713901231813 SideTurn: 1.0385958939647206
Bernoulli Naive Bayes Top 5 Important Feature Weights: Laugh: -3.871201010907891 backHead: -3.871201010907891 otherGaze: -3.871201010907891 downR: -3.871201010907891 gazeUp: -3.1780538303479458	**Bernoulli Naive Bayes Top 5 Important Feature Weights:** Laugh: -3.1780538303479458 backHead: -3.871201010907891 SideTurn: -3.1780538303479458 sideTilt: -3.1780538303479458 waggle: -3.1780538303479458
Linear Discriminant Analysis Top 5 Important Feature Weights: Frown: -8.411073909568955 lipsProtruded: -9.590493508997534 downR: 6.761108001743267 complexHandM: -5.727857205672619 waggle: 4.828417407743949	**LinearDiscriminant AnalysisTop 5 Important Feature Weights:** Frown: -3.0112144669410004 Laugh: -6.901247576631349 gazeUp: -3.1756225463363967 SideTurn: 3.0183750952839645 Close-R: 2.8980736408531627

Before Dropping Features

Refer to Figure 1 on Page 2

After Dropping Features



As described using the table to the right, we can see the changes in distribution. In Experiment #2, we conducted a comprehensive feature importance analysis by comparing the top feature weights before and after dropping selected features. The removal of specific features led to adjustments in the importance of certain cues for deception detection. Notably, features like "Close-R" and "SideTurn" remained consistently important across models, suggesting their relevance in capturing deceptive behavior.

In this second part of Experiment #2, we will understand how these feature weights have impacted model training performance before & after dropping certain features. Let's analyze the following:

Logistic Regression: Accuracy increased from 45.83% to 58.33%, indicating that the feature drop led to better model performance.

Naive Bayes: Accuracy increased from 70.83% to 76.67%, suggesting a marginal improvement after the feature drop.

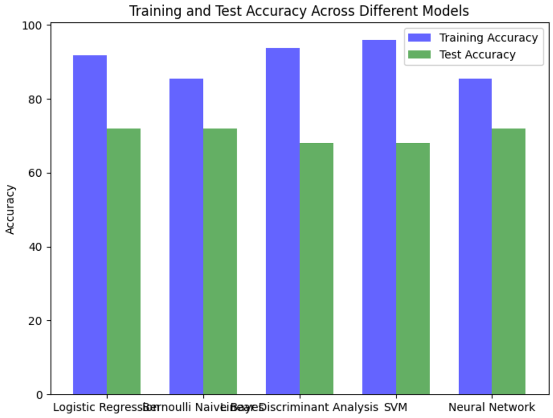
Linear Discriminant Analysis: Accuracy increased, going from 64.17% to 68.17%.

SVM: Accuracy decreased from 66.67% to 64.17%. Despite not having feature weights to analyze, it's clear that SVM's did not benefit from dropping these features

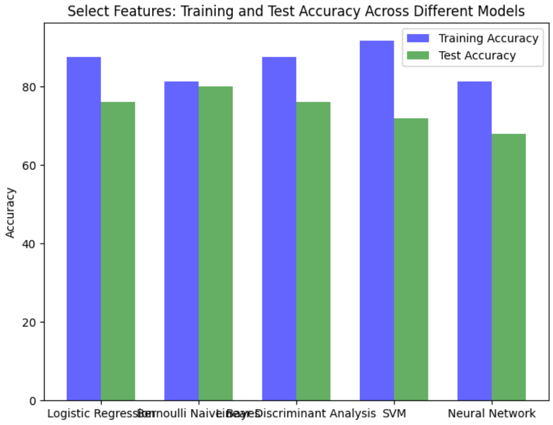
Neural Network: Accuracy remained constant at 62.50%. Despite not having feature weights to analyze, it's clear that NN's did not benefit from dropping these features.

Here's an additional comparison depicting the training and validation accuracy of the training datasets for all 5 of our models:

Before Dropping Features:



After Dropping Features:



We can see that after dropping the features, the overfitting significantly reduced since we lowered the overall complexity of the model by reducing the feature space.

Discussion

In our prior work, we focused on the dataset from "Deception Detection using Real-life Trial Data"[3]. We observed that Linear Discriminant Analysis slightly outperformed the other two models, but all models performed fairly well especially given the small size of the data source. For Linear Discriminant Analysis, the training accuracy was above 90% while the validation accuracy was above 70%, showing signs of some overfitting. We next decided to analyze the features that were most prevalent for each model. Based on the observations, we were able to determine that some features were not as useful in determining if the court footage was deception or not. For example, "gazeDown", meaning looking down, was not a strong indicator for any of the models. However, all

the models significantly used “Raise”, meaning raised eyebrows, as an indication of deception.
The Naive Bayes model emerged as the most effective, achieving an accuracy of 66.67%, suggesting its superior capability in deciphering the subtle nuances associated with bluffing expressions. The other models, while showing competencies, presented relatively similar accuracies, ranging between 54% to 58%. This outcome was analyzed against potential errors and biases, ensuring the robustness of the results. The setup, designed with replicability in mind, provides a comprehensive blueprint for future studies in similar domains.

Our experiments centered on the development of a bluff detection system, applying computer vision classification techniques to analyze players' facial expressions in poker and similar high-stakes situations. We built upon foundational studies in deception detection, such as "Detecting deception through facial expressions" and "Deception Detection using Real-life Trial Data," by translating their findings into practical applications.

Our methodological approach involved using a dataset focused on courtroom gestures and facial expressions, testing various classification models such as Logistic Regression, Bernoulli Naive Bayes, and Linear Discriminant Analysis. We found that Naive Bayes significantly outperformed other models in our testing dataset, although it’s possible that our model oversimplified intricate relationships in our small dataset. By analyzing our model weights, through feature analysis, we’ve discovered the varying importance of gestures, with raised eyebrows ('Raise') being a significant indicator of deception across models.

Considering the limitations and insights from these models, we also explored more complex models like Support Vector Machines (SVMs) and Neural Networks (NNs). SVMs demonstrated a remarkable ability to handle non-linear patterns and outliers, achieving a 66.67% accuracy. However, we observed that in smaller datasets, SVMs might oversimplify relationships, a potential limitation in larger datasets. The Neural Network, using a multi-layer perceptron with a sigmoid activation function, showed a 62.65% accuracy. This was slightly lower than expected, likely due to the small size of our training set, underscoring NNs' preference for larger datasets to capture intricate patterns. Given a larger sample size, we strongly feel that the Neural Network would better capture the hidden patterns amongst our independent categorical features.

Conclusion

In conclusion, this study highlights the potential and challenges of using machine learning for bluff detection in poker games. Our findings suggest that while simpler models like Naive Bayes can be surprisingly effective, the complexity of more advanced models like SVMs and NNs might offer additional benefits, particularly in larger datasets. These insights pave the way for further research in this area, suggesting the exploration of richer datasets and real-time application potential. The project thus contributes a new dimension to the ongoing conversation about the intersection of machine learning, computer vision, and psychological analysis in understanding and predicting human behavior in high-stakes scenarios.

Our results contribute to the field of deception detection by demonstrating the applicability of various machine learning models in a high-stakes context like poker. The experiments revealed the nuanced capabilities of different models in detecting deception and the importance of feature selection in enhancing model accuracy. Compared to previous work, our study extends the understanding of machine learning's role in psychological analysis, particularly in environments where non-verbal cues are crucial.

Acknowledgements

We extend our profound gratitude to Professor Kuleshov for his invaluable guidance and insightful teachings throughout the duration of this course.

References

[1] Feinland, J., Barkovitch, J., Lee, D., Kaforey, A., Ciftci, U.A. (2022). Poker Bluff Detection Dataset Based on Facial Analysis. In: Sclaroff, S., Distante, C., Leo, M., Farinella, G.M., Tombari, F. (eds) Image Analysis and Processing – ICIAP 2022. ICIAP 2022. Lecture Notes in Computer Science, vol 13233. Springer, Cham. https://doi.org/10.1007/978-3-031-06433-3_34

[2] Merylin Monaro, Stéphanie Maldera, Cristina Scarpazza, Giuseppe Sartori, Nicolò Navarin, Detecting deception through facial expressions in a dataset of videotaped interviews: A comparison between human judges and machine learning models, Computers in Human Behavior, Volume 127, 2022, 107063, ISSN 0747-5632, https://doi.org/10.1016/j.chb.2021.107063

[3] Verónica Pérez-Rosas, Mohamed Abouelenien, Rada Mihalcea, and Mihai Burzo. 2015. Deception Detection using Real-life Trial Data. In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction (ICMI '15). Association for Computing Machinery, New York, NY, USA, 59–66.

https://doi.org/10.1145/2818346.2820758

[4] D. Vinkemeier, M. Valstar and J. Gratch, "Predicting Folds in Poker Using Action Unit Detectors and Decision Trees," 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xi'an, China, 2018, pp. 504-511, doi: 10.1109/FG.2018.00081. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8373874

[5] PokerGO. "Celebrities Square off in the $50,000 Enclave Poker Invitational in Las Vegas!" YouTube, uploaded by PokerGO, 29 Aug 2023, https://www.youtube.com/watch?v=nurPvqOkp_A

[6] PartyPokerTV. "Premier League Poker S4 EP19 | Full Episode | Tournament Poker | partypoker" YouTube, uploaded by PartyPokerTV, 18 Feb 2020, https://www.youtube.com/watch?v=lunRYGW9Khw