An analysis of Balloon Analogue Risk Task

Abtin Mogharabinng
Statistics Department
Middle East Technical University
Ankara, Turkey
abtin.mogharabin@gmail.com

***Abstract*—The Balloon Analogue Risk Task (BART) is a widely recognized cognitive science tool for evaluating risk-taking behavior. This project leverages BART data from an experiment designed to investigate the impact of alcohol consumption on risk-taking. Participants, subjected to varying doses of alcohol, engaged in three blocks of trials with different probabilities of the balloon bursting, presented in a randomized order. Each block consisted of multiple trials, and participants were informed of the burst probability beforehand. Our analysis aims to explore the factors influencing risk-taking behavior and rewards using a combination of statistical tests and machine learning models, including XGBoost, LightGBM, regression, neural networks, and random forests. The findings are expected to enhance our understanding of how alcohol affects decision-making processes related to risk.**

Keywords— cognitive science, balloon analogue risk task, statistical modeling, machine learning

I. INTRODUCTION

Risk-taking behavior is a fundamental aspect of human decision-making, influencing various domains such as finance, health, and social interactions. BART is a popular experimental tool for assessing this behavior. In BART, participants inflate a virtual balloon to earn rewards, with each pump increasing the risk of the balloon bursting and losing the accumulated rewards. The decision to continue inflating or to bank the earnings reflects the participant's risk propensity.

This project focuses on an experiment that examines the influence of alcohol consumption on risk-taking behavior using BART. The within-subjects design involved participants consuming different doses of alcohol—sober, tipsy, and drunk—before completing the task. Each participant experienced three blocks of 30 trials each, with varying burst probabilities (0.1, 0.15, 0.2) presented in random order. Participants were informed of the burst probability before each block, and the reward per pump was a percentage of the money earned so far. The experiment data is structured in name-id.txt files for each participant, with 90 rows of data per file, corresponding to the three conditions and three burst probabilities. The randomized relationship between sessions and conditions is documented in a provided mapping matrix.

In this project, we will analyze the collected data to identify the elements that influence risk-taking behavior and rewards under different alcohol conditions. We will employ statistical tests to examine the effects of alcohol on risk-taking and use various machine learning models to predict risk-taking behavior and reward outcomes.

II. DATA PREPROCESSING

The first step of the project is to combine all the data files into a single, organized data frame. In this step, I extracted the data of different patients from their corresponding txt files. Then I organized all the information provided in the data and generated the following 10 variables:

- **participant**: identifies the participant. Although it's numerical, it's essentially categorical since it represents different people.
- **condition**: represents the condition of each participant in each session
- **p_burst**: represents the probability of a balloon bursting in each session
- **trial**: represents the trial number of each participant in each session
- **pumps**: the number of times a balloon is pumped for each participant in each point of a session
- **cash**: the amount of money gained in each trial
- **total**: total reward of a block
- **session**: an integer representing a session (1, 2, or 3
- **block**: indicates which block the trial belongs to

These 10 variables are the key variables used to analyze the data in the EDA and the descriptive statistics sections. In the statistical modeling section, to better extract the key features of the data, I used these variables and generated 4 new numeric variables which provide us some key information that was not included in the previous models:

- **NOP**: mean number of pumps across trials, within a block
- **pC**: proportion of cashed trials in a block
- **pE**: proportion of explosions in a block with respect to the total number of pumps in a block

III. LITERATURE REVIEW

When individuals take risks, they seek potential rewards while exposing themselves to various dangers such as financial ruin, cocaine addiction, sexually transmitted infections, and even death. A significant factor influencing why some people take risks while others choose caution is substance abuse (Adlaf & Smart, 1983), with alcohol being a common substance of abuse.

The effects of alcohol on risk-taking have been widely researched. Alcohol abuse has been found to increase risk-taking behavior during driving (Burian, Liguori, & Robinson, 2002), reduce the perceived negative consequences of risky actions, and elevate participation in unsafe sexual activities (Kalichman, Heckman, and Kelly, 1996). Additionally, alcohol consumption has been linked to an increased number of accidents (Cherpitel, 1993).

In this project, I examined the impact of three different alcohol doses and several other relevant elements on the risk-taking behavior of participants in the BART cognitive test.

IV. MISSSINGNESS

Considering that my data does not have any missing data, I generate them by myself. For this purpose, I randomly deleted 5% of the data from each variable. After this, I conducted the EDA and CDA analysis without replacing any of my missing values. After the EDA and CDA sections were finished, I then decided on a suitable imputation method to restore the missing values. We already know that we are dealing with a missing completely at random (MCAR) situation because I randomly generated these missing values. In this step, I used Predictive Mean Matching for numerical variables and Polynomial Regression imputation for factors. The reason I didn't try Single Imputation is that my data was specifically designed with different participants and different conditions of participants in mind. These differences are critical to have the best analysis of the data later. If we just use a Single Imputation and replace the missing values of all variables with the same mean/median values, these differences would be

lost. At the same time, NOCB and LOCF were not suitable because they would have resulted in the data of different participants and different conditions being mixed.
Now, the data is ready to be used for EDA and modeling. Before we start the EDA, we also take a look at the data's structure and ensure that all key variables are preprocessed in a suitable format:

```
[1] "The final version of the data:"
'data.frame': 4860 obs. of 10 variables:
$ participant: Factor w/ 18 levels "1","2","3","4",..: 14 1
1 5 1 1 1 1 1 1 ...
$ condition : Factor w/ 3 levels "drunk","sober",..: 1 2 1 1
1 1 1 1 1 1 ...
$ p_burst : Factor w/ 3 levels "0.1","0.15","0.2": 2 2 2 2 1
2 2 2 2 2 ...
$ trial : Factor w/ 30 levels "1","2","3","4",..: 7 2 3 4 5 1
7 8 16 10 ...
$ pumps : num 12 1 4 2 6 5 14 5 2 5 ...
$ cash : num 0 0 0 0.81 0 0.95 2.23 0 0 0.95 ...
$ total : num 3.2 0 0 0.81 0.81 0 4.33 2.57 5.3 3.52 ...
$ session : Factor w/ 3 levels "1","2","3": 1 3 1 1 1 1 1 1
2 2 ...
$ block : Factor w/ 3 levels "1","2","3": 2 2 2 3 2 2 2 2 2
2 ...
$ explosion : logi TRUE TRUE TRUE FALSE TRUE
FALSE ...
```

V. Descriptive Statistics

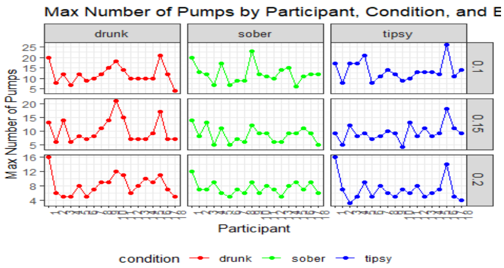
For this dataset, I will not provide explicit information on the descriptive statistics here. As mentioned in the previous section, this dataset was developed through careful considerations and strict experimental design. Because of this, the data creators have already ensured that the distribution of the values for the participant, condition, p_burst, trial, session, and block variables were balanced throughout the dataset. The key variable that needs descriptives here is the explosion variable. I specifically checked the frequency values of this and provided a frequency table for the explosion variable across different labels. There are around 2300 exploded balloons and around 2500 non-exploded balloons. Overall, there was no imbalance problems with this variable. Even after I stratified for different p_burst values.

VI. EDA and CDA

Now we visualize and apply suitable statistical tests to analyze the data. The focus of my analysis would be the following research question:

- How does the participants' condition affect their maximum risk-taking (maximum number of times that participants pump a balloon)?
- How does the participants' condition affect their average risk-taking?
- How does the participants' prior on the burst probability of balloons affect their maximum risk-taking?
- How does the participants' prior on the burst probability of balloons affect their average risk-taking?
- What about the pumps that lead to explosions? Does increased risk-taking in low burst probabilities result in a significantly higher number of explosions?
- Do different burst probabilities have any effects on total rewards?
- Do different conditions have any effects on total rewards?
- How do individual participants differ in their mean total rewards across trials?
- How do individual participants differ in their mean number of pumps in different conditions?

I start EDA with the following figure:



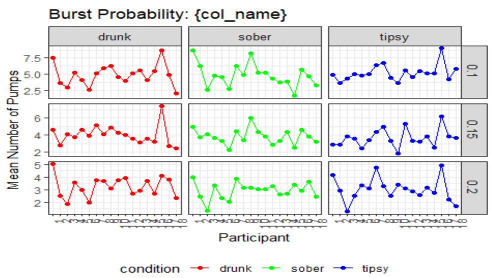
We can observe a general trend here: although different alcohol conditions don't have much of an effect on the maximum risk-taking of participants, the differences in the probability of burst seem to have a rather significant effect. We can see that a higher probability of burst values has resulted from participants having a smaller maximum pump number. We check this using a two-way ANOVA test. Note that by our design of the experiment, we already know that independence and homogeneity assumptions are met. I also checked the normality assumption and our residuals are approximately normal.

```
##                    Df Sum Sq Mean Sq F value   Pr(>F)    
## p_burst             2  702.9   351.4  26.431 1.38e-10 ***
## condition           2   16.9     8.5   0.636    0.531    
## p_burst:condition   4   55.7    13.9   1.048    0.385    
## Residuals         153 2034.3    13.3                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value is extremely small. It is clear that we can reject the null hypothesis and conclude a significant effect from burst probability on the maximum number of pumps. While the condition has a big p-value and we fail to find any significant effect from the condition to the maximum risk-taking of participants. And the result of the Tukey multiple comparisons of means confirms that as burst probability increases the maximum pump number decreases.

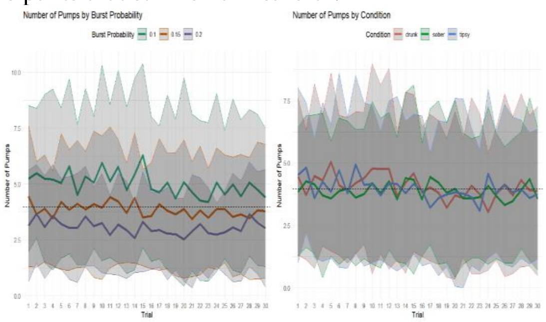
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = max_pumps ~ p_burst * condition, data =
max_pumps_df)
##
## $p_burst
##               diff       lwr        upr     p adj
## 0.15-0.1 -3.222222 -4.883089 -1.5613551 0.0000271
## 0.2-0.1  -5.037037 -6.697904 -3.3761699 0.0000000
## 0.2-0.15 -1.814815 -3.475682 -0.1539477 0.0285203
```

Now we check if the average number of pumps is also as significantly affected as the maximum number of pumps. Especially, we need to check if the condition of participants shows any significant effect on their average risk-taking.



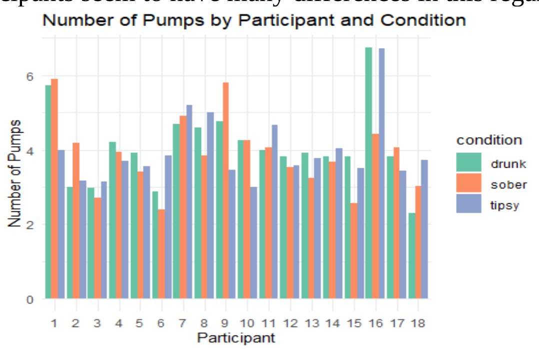
Similar to the previous part, we observe an effect from burst probability but no effect from the condition. We use another two-way ANOVA text and check this. We again observe a significant effect from p_burst (p-value 7.33e-14) but no impact from condition (p-value 0.710). I also applied Tukey's multiple comparisons of means which shows us that higher burst probabilities indeed decrease participants' average risk-taking.

The following 2 plots allow us to further confirm the effect of burst probability on risk-taking. But we also observe a new observation: although lower burst probabilities show a higher risk-taking, they also show a higher variance. Indicating that although in low burst probabilities, some participants take a lot of risks, but there are also some participants that still remain careful.



To test this, we apply Levene's Test and observe a significant difference in the variances of the mean number of pumps across different burst probabilities. The p-value is extremely small (2.2e-16) which shows that the null hypothesis (that the variances are equal) can be rejected with high confidence.

We previously observed the surprising result that the condition of participants, in general, doesn't have an effect on the number of pumps. Now, we analyze this across different participants. In the following plot, we observe that different participants seem to have many differences in this regard:

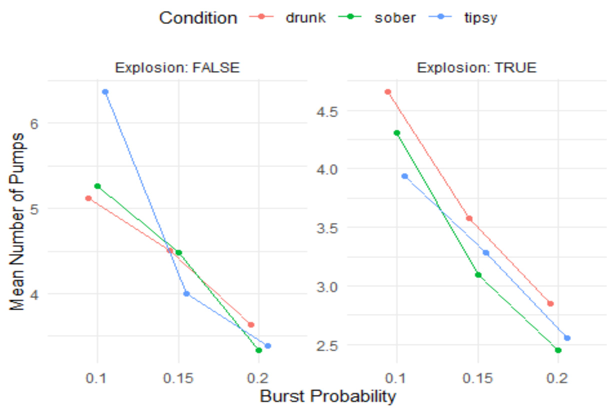


This shows the reason behind the surprising conclusion we had on the effect of condition on pumps before. It seems the reason we failed to find a significant effect before is that alcohol (condition) has vastly different effects on different participants. It increases the risk-taking of some participants, decreases the risk-taking of another group of participants, and doesn't affect some participants. This is why testing the grand average between all participants gave us the conclusion that "condition doesn't affect risk-taking"! Because the effect of alcohol on different participants could be the total opposite of each other. Now, we conduct independent ANOVA tests for different participants and test this. Again, we make sure that all ANOVA assumptions are met.

```
## Participant 1 : Significant effect (p-value = 0.003618426 )
## Participant 2 : Significant effect (p-value = 0.0003660177 )
## Participant 3 : No significant effect (p-value = 0.6299202 )
## Participant 4 : No significant effect (p-value = 0.3193875 )
## Participant 5 : No significant effect (p-value = 0.4583926 )
## Participant 6 : Significant effect (p-value = 2.499767e-07 )
## Participant 7 : No significant effect (p-value = 0.4572935 )
## Participant 8 : Significant effect (p-value = 0.01549591 )
## Participant 9 : Significant effect (p-value = 6.260573e-05 )
## Participant 10 : Significant effect (p-value = 0.007253695 )
## Participant 11 : No significant effect (p-value = 0.209985 )
## Participant 12 : No significant effect (p-value = 0.6446015 )
## Participant 13 : No significant effect (p-value = 0.2116296 )
## Participant 14 : No significant effect (p-value = 0.5574186 )
## Participant 15 : Significant effect (p-value = 0.001041693 )
## Participant 16 : Significant effect (p-value = 0.0009535787 )
## Participant 17 : No significant effect (p-value = 0.303128 )
```

Now the results sound more logical. We observe that alcohol has no significant effect on the number of pumps of 9 participants. But it has significant effects on the other 8 participants.

Now, we check if different burst probabilities have a significant effect on the number of explosions.

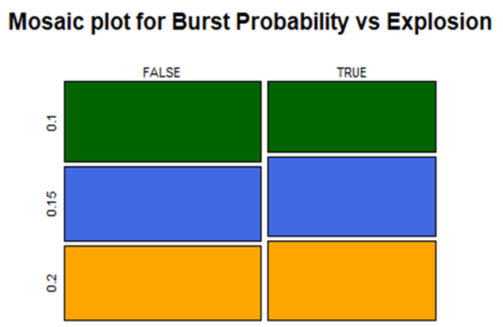


We observe a clear connection between them. We also observe that the higher risk average pump number comes from 0.1 burst probability from the cases that balloon has not exploded. For better p_burst vs explosion interpretation, we check the frequency table and odds ratio.

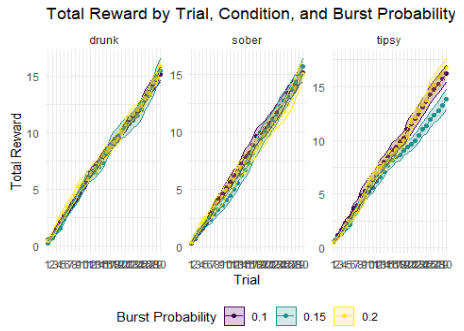
```
##         FALSE  TRUE
##   0.1     929   691
##   0.15    843   777
##   0.2     852   768
## $estimator
## [1] 1.239171
##
## $SE
## [1] 0.08644442
##
## $conf.interval
## [1] 1.046043 1.467955
##
## $conf.level
## [1] 0.95
```

We can observe that there is a statistically significant association p_burst and explosion. The odds of an explosion are lower in the higher p_burst values. The chi-square test of independence also confirms the dependence of explosion on p_burst (with a p-value of 0.003879).

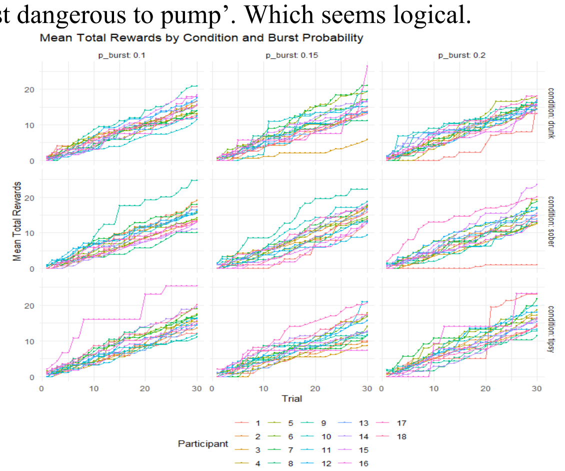
Now we check their mosaic plot. We can observe that



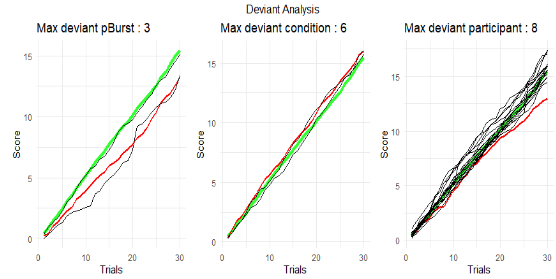
burst probability 0.1 has the smallest rate of explosion and the largest rate of not exploding. This shows us that participants seems to have a good risk-taking in general and they don't get overly reckless in p_burst=0.1 just because they think there is a lower chance of explosion.



Now we check the differences between our participants in their process of gaining rewards. We observe a more stable increase in the rewards of our participants in the drunk condition compared to other conditions. While we observe the most variations for p_burst 0.1 and 0.2, where the participants think the balloons are the 'safest to pump' or 'the most dangerous to pump'. Which seems logical.



Now, we plot three line plots to do a simple deviant analysis of our key variables. The following plot visualizes the trial-score vectors for three different variables: burst probability, condition, and participant. We identify the most deviant values within each variable type based on the mean squared deviation from the grand mean. The red line represents the trial-score vector for the most deviant value within the variable type, the green line represents the grand mean across all trials for the variable being analyzed, and the black lines show us the remaining (non-most-deviant) labels of each variable.



Finally, we check the correlations that are higher than a threshold using suitable correlation method for every 2 variables:

```
p_burst and pumps correlation is -0.2768
cash and explosion correlation is -0.849
pumps and explosion correlation is -0.1938
pumps and block correlation is -0.2474
pump and cash correlation is 0.4752
trial and total correlation is 0.8911
p_burst and block correlation is 0.7865
```

It seems we have some correlation problems we need to deal with in our modeling section. In addition, the source that shared the data explicitly mentioned that this data has large problems with interaction between variables.

VII. Statistical Modeling

In this section, I apply a number of basic statistical models to my dataset. The models are as follows:

- Classification: Burst Prediction
- Classification: Probability of Burst Prediction
- Regression: Predicting the Trial's Reward

For each model, I made sure to check all assumptions of the corresponding model. Then I applied 5-fold cross-validation and fine-tuned all my models across different hyperparameters. In the end, I reported the suitable metrics for my predictions. The trained models were: Support Vector Machines or Support Vector Regression, Neural Networks, XGBoost, LightGBM, Random Forest, and Logistic Regression or Multiple Linear Regression. In the end, I saved the suitable metrics and compared these models with each other.

Considering that I trained and tuned over 15 different models, it would take up a lot of space to explain each one individually. Therefore, I will first explain the strategies used to train all of my models. Then, I will report the results of the different models for each task and analyze the results of only the best models.

1. Multiple Linear Regression (MLR) / Logistic Regression

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. On the other hand, MLR is a statistical model that trains a linear combination of multiple variables to predict a continuous variable.

First, I applied a basic regression model to my data. The type (Logistic/MLR) was dependent on whether I was modeling a regression or a classification problem. Then, the corresponding assumptions of the model were checked to ensure the model did not violate any assumptions. For logistic regression, these assumptions were linearity of the logit, absence of multicollinearity, and independence of errors. For MLR (Multiple Linear Regression), these assumptions were linearity, independence, homoscedasticity, and normality of residuals. Note that some of these conditions (such as the independence of most of our independent variables), could be assumed to be mostly satisfied because of our strict design of experiment and the most challenging MLR assumption was the normality assumption. In cases where some of the assumptions were not met, I applied suitable transformations such as normalization, square transformation, and Box-Cox transformation. I also added suitable interaction terms to the model to address confounding issues.

2. Support Vector Machines (SVM)

Support Vector Machine is a supervised learning algorithm that can be used for both classification and regression problems.

Similar to the previous section, I decided on a suitable modeling approach based on my tasks. For classification, SVM was used, and for the regression problem, SVR was used. First, I applied cross-validation and tuned my models for the best cost and gamma hyperparameters. Then, I trained the best model on my whole train set and calculated the suitable classification metrics on the test set.

3. XGBoost

XGBoost (Extreme Gradient Boosting) is an optimized implementation of gradient boosting that builds an ensemble of decision trees sequentially. It minimizes a loss function by adding new trees that correct the errors made by the previous ones, and it employs techniques like regularization, parallel processing, and tree pruning to improve performance and prevent overfitting.

To train this model, I employed a wide grid search strategy. The model was tuned for several hyperparameters, including the number of boosting rounds, learning rate, max depth of trees, minimum loss reduction required for further partitioning, subsample ratio of columns when constructing each tree, minimum sum of instance weight (hessian) needed in a child, and subsample percentage of the training instances.

4. Artificial Neural Networks

Artificial Neural Networks (ANNs) are a simple imitation of the human brain. They consist of interconnected layers of nodes (neurons) that process input data to generate an output. ANNs learn by adjusting specific weights through a process called backpropagation, which minimizes the error between the predicted and actual outputs by iteratively updating weights using gradient descent through multiple layers of neurons.

In my case, I tuned each of my ANN models for a variety of different hidden layer numbers and different numbers of neurons in each hidden layer.

5. Random Forest

Random Forest works by constructing multiple decision trees during training and outputting the mode of the classes (classification) or mean prediction (regression) of the individual trees. It reduces overfitting by averaging multiple trees, which are built using different subsets of the training data and features, leading to more robust and accurate predictions.

For my task, I mainly tuned for the Number of trees, the minimum size of terminal nodes, the maximum number of terminal nodes, and the number of features to consider when looking for the best split. I focused on these because I noticed the most significant effect of these hyperparameters on the result of prediction in my data.

VIII. Main Tasks

1. Burst Prediction

Here, I tried to predict if a balloon was going to explode depending on different elements. I began with the basic formula ` explosion ~ condition + p_burst + pumps + session + participant` on a simple logistic regression problem. The resulting logistic regression model had many problems. None of the variables except pumps and participants seem to have any significance. This was likely due to the interaction between our independent variables.

To deal with this problem, I applied a square transformation to p_burst, encoded the session variable, and kept only the information of the third session (because I found this session to be the most relevant after different experiments with different models), normalized the pump variable, added suitable interaction terms to the model. In the end, I used a suitable encoding method for the categorical variables that needed encoding.

```
##
## Call:
## glm(formula = explosion ~ condition_pb * session3 + pb * pumps +
##     pb + participant, family = "binomial", data = final_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2876  -0.9963  -0.6572   1.1065   2.9856
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)              2.0311     0.1979  10.265  < 2e-16 ***
## condition_pb            -5.1447     1.5795  -3.257 0.001125 **
## session3                -0.3699     0.1252  -2.953 0.003142 **
## pb                      13.4461     5.2621   2.555 0.010610 *
## pumps                   -0.1441     0.0228  -6.319 2.63e-10 ***
## participant2            -1.6937     0.1961  -8.635  < 2e-16 ***
## participant3            -2.2576     0.2025 -11.151  < 2e-16 ***
## participant4            -1.4679     0.1934  -7.588 3.24e-14 ***
## participant5            -1.7561     0.1971  -8.909  < 2e-16 ***
## participant6            -2.4159     0.2044 -11.817  < 2e-16 ***
## participant7            -0.7599     0.1936  -3.926 8.64e-05 ***
## ……… (the rest of the output was removed to reduce the space)
```

We observe a quite good model now. Not only does the deviance show a good performance for the model, but all our coefficients also seem significant. Deviance Residuals also show a relatively good performance. In addition, in our coefficients, we observe that the pump has a negative effect on the explosion, which is a logical interpretation. Also, we observe that as the probability of burst increases, the chance of an explosion also increases. VIF values are also not that big. The only element to note is that condition_pb has a low effect alone. But it has a high positive effect in its interaction term with session. The model seems logical and does not contradict our results in the EDA section.

Now that I generated a good combination of different transformations and interaction terms, I used the same equation to finetune and train the rest of my models. In the next section of this report, we will compare the performances of different models and analyze the best model.

2. Probability of burst prediction

In this model, we try to predict if the burst probability of a balloon is 0.1 or 0.2. Similar to the previous mode, we use the interaction effects in the model. We also normalize the pump variable. I did not print the summary of the model because of the high number of variables (1 whole page). The results are available in the uploaded R file. The model followed the equation ` p_burst ~ explosion + condition + pumps_n * explosion * participant + cash`. P-values were mostly significant except for a small number of labels (dummy variables) and the resulting coefficients looked logical. The highest effect in the burst probability prediction came from the variables: explosion, pumps, the amount of cash, and the effect of each participant. All of their effects followed our results in the EDA section.

I again used this refined formula for the rest of my models and comprehensively fine-tuned them. The results of the exact classification metric results will be analyzed in the next section.

3. Total reward prediction

In this section, I tried to predict the total reward gained by each participant in different conditions during a complete block. To do this, used variables that combined all results of each block (every 30 trials). The main problem here was that the normality assumption of linear regression was not met. For this reason, I applied box-cox transformation. I also dealt with the interaction problem by adding suitable interaction terms. I also tuned the model for elastic net regularization to deal with some of the remaining multicollinearity. By the end, the equation of the regression model that worked the best was `Total_reward ~ condition + pumps_sum + pE + mNOP + NOE + p_burst + participant`. Where the summations were taken over the whole block for different participants.

The coefficients also seemed logical. We observed that the sober condition has a positive effect on the total reward won. However, the effects of tipsy and drunk were similar to each other and both had slight negative effects. A similar effect was also observed in the EDA section where the effect of the condition had to be stratified for participants so that we can observe more significant effects. I also observed that each participant's dummy variable had a significant effect on their total reward. NOP (mean number of pumps across trials) also had a small effect on the reward. This makes sense because although more pumps would increase the chance of getting more rewards, it also increases the chance of a balloon exploding.

After deciding on the best data format, I then trained and fine-tuned the rest of the SVM, NN, RF, and XGBoost models. We compare and analyze the best results in the next section.

IX. Results

For Classification tasks, the metrics reported in this section are accuracy, sensitivity, specificity, NPV, and PPV. These values are calculated according to the formulas in figure 1.

	True Class: Positive	True Class: Negative	Measures
Predicted Class: Positive	True Positive **TP**	False Positive **FP**	Positive Predictive Value (PPV) $\frac{TP}{TP+FP}$
Predicted Class: Negative	False Negative **FN**	Ture Negative **TN**	Negative Predictive Value (NPV) $\frac{TN}{FN+TN}$
Measures	Sensitivity $\frac{TP}{TP+FN}$	Specificity $\frac{TN}{FP+TN}$	Accuracy $\frac{TP+TN}{TP+FP+FN+TN}$

(1)

For regression tasks, on the other hand, the RMSE results of each model are reported. RMSE is calculated following the equation provided in figure 2.

$$RMSE=\sqrt{\frac{\sum_{i=1}^{N}(Predicted_i-Actual_i)^2}{N}} \qquad (2)$$

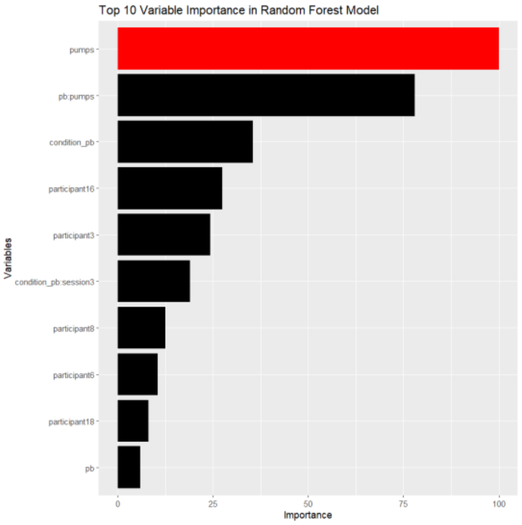
Now, we observe the performance of our models for each task. The top 2 performances for each metric are highlighted.

1. Burst Prediction

	ACC	SN	SP	PPV	NPV
LR	0.65	**0.77**	0.53	0.66	0.66
SVM	0.65	**0.80**	0.49	0.65	0.68
ANN	0.67	0.71	**0.64**	0.69	0.65
RF	**0.69**	0.75	**0.62**	**0.70**	**0.69**
XGBoost	**0.70**	0.77	0.61	**0.70**	**0.70**

Table 1 Performance Comparison

It is obvious that random forest is the best model for burst prediction. Not only does RF show the best ACC, SP, PPV, and NPV results, the SN is also not much lower than the other models. After fine-tuning, this model was shown to provide the best result after I used 4 features to consider when looking for the best split. Below, we observe the most important variables in the best prediction results of the RF model.



The most significant parameters are shown to be the number of pumps, the interaction effect of participants prior on the burst probability, and the interaction effect of burst probability and condition. All of which seem logical.

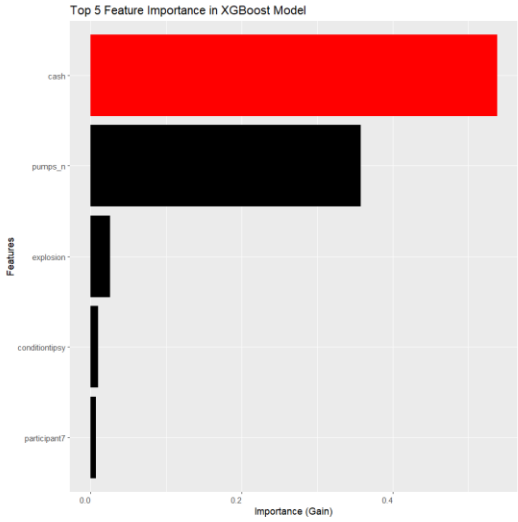
Now, we move to the second classification task.

2. Probability of burst prediction

	ACC	SN	SP	PPV	NPV
LR	0.76	0.67	**0.86**	**0.83**	0.72
SVM	0.75	**0.87**	0.63	**0.83**	0.70
ANN	0.62	**0.79**	0.44	0.58	0.68
RF	**0.78**	0.73	0.83	0.81	**0.75**
XGBoost	**0.78**	0.68	**0.88**	**0.85**	**0.73**

Table 2 Performance Comparison

Clearly, XGBoost shows superior overall performance for predicting the burst probability, albeit having some rather significant problems in SN. Meaing that our model has some problems in predicting the cases where burst probability is low (0.1). After tuning, the best XGBoost hyperparameters were determined as follows: the number of boosting rounds (nrounds) was set to 150, the maximum depth of the trees (max_depth) was set to 9, the learning rate (eta) was set to 0.01, the minimum loss reduction required for a further partition (gamma) was set to 0, the subsample ratio of columns when constructing each tree (colsample_bytree) was set to 1, the minimum sum of instance weight needed in a child (min_child_weight) was set to 1, and the subsample percentage of the training instances (subsample) was set to 0.8. We can observe the top 5 variable effect on the predictions in the following figure.



We observe that the amount of cash earned by participants and the number of pumps in trials of a block heavily effects the burst probability's prediction. This is interesting because it shows that the participants were not able to suitably react to the changes in the burst probability. Remember that participants were informed of the burst probability before each block. Because of this, it would have been expected from them to robustly think about the burst probability and pump the balloon for a suitable number of times. The fact that the burst probability's prediction was affected so much by the cash and pump variables shows that our participants were heavily affected by their prior knowledge of burst probability.

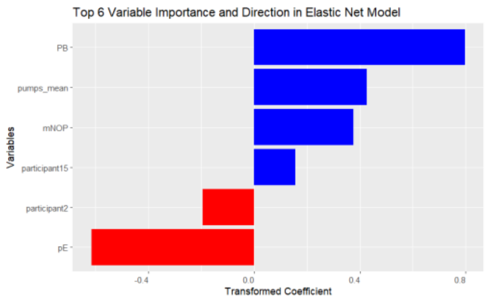
Unfortunately, the generated tree by the XGBoost model was huge. Because of this, I am not able to provide the tree here but you can easily observe the tree in the corresponding R codes of this report.

3. Total reward prediction

	RMSE
MLR	0.5074
SVR	0.9204
ANN	0.7377
RF	0.6548
XGBoost	0.5497

Table 3 Performance Comparison

We observe that the regression model with elastic net gives the lowest error for this model. This model was tuned for different regularization parameters. The best hyperparameters used in the end were alpha = 0.6 and lambda = 0.02154435. In the following figure, we can observe the top 6 most important variables in the prediction of total reward won in a single block.



In this figure, we observe the most significant effect from the burst probability variable. This shows that participants are heavily affected by their prior on the probability of a balloon. Because of this as the probability of burst, they get more careful in their actions and can act logically and get more rewards. On the other hand we observe that as the proportion of explosions with respect to the number of pumps increases, the participants win less reward. This (pE variable) shows us the effect of getting careless and pumping too much, which would result in an opposite effect to the cases where the participants pumps but not too much (this could be observed when we subtract pE from pumps_means).

Note all of these results are very similar to what we observed in the XGBoost model previous section.

X. Conclusion

In this study, we examined how alcohol consumption and other relevant elements affect risk-taking behavior using the Balloon Analogue Risk Task (BART). We used statistical tests and machine learning models like XGBoost, Random Forest, Elastic Net, SVM, and Neural Nets to model and analyze the data. Our results showed that the probability of the balloon bursting significantly impacts risk-taking, with higher probabilities leading to fewer pumps. While the overall effect of alcohol on risk-taking was not significant, individual analysis revealed that alcohol affects participants differently, increasing risk-taking in some and decreasing it in others.

The Random Forest model was the best at predicting balloon bursts, with the number of pumps and interactions between conditions and burst probabilities being key factors. For predicting burst probabilities, XGBoost performed the best, with the amount of cash earned and the number of pumps as significant predictors. The Elastic Net model had the lowest error in predicting total rewards, highlighting the impact of burst probability and the proportion of explosions.

References

[1] Adlaf, E. M., & Smart, R. G. (1983). Risk-taking and drug-use behaviour: an examination. Drug and Alcohol Dependence, 11, 287–296.

[2] Burian, S. E., Liguori, A., & Robinson, J. H. (2002). Effects of alcohol on risk-taking during simulated driving. Human Psychopharmacology, 17, 141–150.

[3] Cherpitel, C. J. (1993). Alcohol and injuries: a review of international emergency room studies. Addiction, 88, 651–665.

[4] Kalichman, S. C., Heckman, T., & Kelly, J. A. (1996). Sensation seeking as an explanation for the association between substance use and HIV-related risky sexual behavior. Archives of Sexual Behavior, 25, 141–154

[5] Van Ravenzwaaij, D., Dutilh, G., & Wagenmakers, E. J. (2011). Cognitive model decomposition of the BART: Assessment and application. Journal of Mathematical Psychology, 55(1), 94-105.