Clustering of player styles in POT LIMIT TEXAS HOLD'EM poker variation

Gediminas Sadaunykas

City, University of London

1. INTRODUCTION

If you are willing to learn poker, chances are you will be told ''it takes 10 minutes to learn, though lifetime to master it''. Texas Hold'em is the most widely played variation.[1] In summary, in poker number of players (between 2 and 10) are performing one of four actions at four different stages of a poker hand[2], with a goal to win as much money/chips as possible. It could be achieved through successful deception or completion of hand better than that of others. One of the simplest versions – Limit Hold 'em Heads Up contains $1.38 * 10^{13}$ information sets. It was solved in 2015 with CFR+ algorithm (Bowling et al., 2015) . Achievement is more astonishing given the fact that magnitude of number of information sets was of three orders larger, than any game solved previously.

The motivation this paper comes from authors' opinion that any solution to games with imperfect information[3] could be further improved via human/machine interaction. The value of clustering poker player styles, in both: computational poker agent modeling as well as real time human decision making. (North, 2009) (Harrington and Robertie, 2004) The lack of comprehensive studies on POT-LIMIT Texas Hold'em variation.

2. DOMAIN DATA AND QUESTIONS

Data was taken from famous University of Alberta poker research group website[4]. Two reasons being: it requires minimal preprocessing before being analyzed and data from the same source had been used in solving Limit Heads Up game (Bowling et al., 2015). It was generated by players competing for fake money over internet relay chat (IRC) channel. Participants were known to be mostly computer science geeks and poker enthusiasts. As a result, overall quality of information is considered to be quite good and applicable to recent times. There are 1 826 766 POT-LIMIT Hold'em hands, played between 1995 and 2001 in total. Data is contained in 4892 files, unique for each player and describing the universe of actions performed, along with outcomes.

Harrington (Harrington and Robertie, 2004) suggested that any poker hand in any given session could be described by ten variables. Two out of ten can never be objectively observed and measured: opponent cards and opponent style. It is argued that by accurately describing opponent style, opponent cards could be deduced from remaining variables. Borer (Borer 2007) popularized three measures[5] to categorize players into 4 clusters. Measures are AF, VPIP and PFR (Appendix A). Clusters: loose-passive, loose-aggressive; tight-passive; tight-aggressive. However, nowadays as a result of huge increase in data and availability of various measures, the significance of these variables is arguable.

Main goal of this paper is to evaluate and expand on Borers' variables, in poker player style clustering. Secondary one, is to deduce the characteristics of a winning poker player.

3. TASKS AND APPROACHES.

Tasks and analysis is divided into three parts – preprocessing; unsupervised clustering; supervised clustering. Main task in preprocessing was to derive unique player characteristics and to reduce data-frame where every row is unique player, instead of hand. Borers' variables had been derived according to (Borer 2007), using regular expressions in python. Expansion variables DECEPT and COMMPROP derived according to authors personal opinion; WTSHWD inspired by being popular statistic in PokerTracker.[6]

K-means[7] was used to cluster players according to specified characteristics. It calculates Euclidean distances; and adjusts centroids iteratively to find data clusters in high dimensional space, is also robust and fast. Results for different combinations of variables had been evaluated using Silhouette coefficient; Calinski-Harabaz score; for full model optimal number of variables derived from Silhouette plot. Results are exported to Tableau for further analysis using parallel coordinate maps (PCM) and distribution histograms. Averages are taken in PCM to reduce clutter; one coordinate – proportion of winning players added next to characteristics for immediate view on profitability of different styles. Frequency histograms, in which every bar divided in four clusters, depicts distribution of profitability for different styles of poker player. Welch t-tests are performed to test the significance of differences in mean profitability, as a result of violation of ANOVA assumptions and Welch –test superiority. (Mooser and Stevens, 1992)

Second part begins with visual analysis of PCMs with average characteristics. The visual methods applied are very abstract and would not uncover non-linear complex relationships, therefore implementation of Random Forrest algorithm follows. (Breinman 2001) Main reason for choosing this particular method, is its

[1] It assumed that reader understands rules of Texas Hold'em, which could be found: https://www.partypoker.com/how-to-play/texas-holdem.html

[2] Glossary of poker terms: https://en.wikipedia.org/wiki/Glossary_of_poker_terms

[3] Imperfect information, stochastic outcomes and non-cooperative behaviour are three main features of every poker variation.

[4] http://poker.cs.ualberta.ca/irc_poker_database.html

[5] AF, VPIP and PFR often referred as *Holy Trinity.*

[6] Popular poker analysis software: https://www.pokertracker.com/products/PT4/

[7] http://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

robustness and predictive power. (Caruana et.al, 2006) However, algorithm requires some tuning and adjustment for results of a model converge to cross-validated result. Classifier is evaluated using various visual techniques – *out-of bag*[8] error plots - to find optimal number of decision trees; learning curves – to investigate the algorithms ability to learn with additional data; ROC-AUC curves and confusion matrices to picture the performance. Feature contributions had been extracted using *treeinterpeter*. [9](Saabas 2014) Contributions compared to implicit measure - *GINI*[10] impurity decrease. Treeinterpreter gives linear sum of feature contributions; while GINI is more versatile measure. Results are exported back to Tableau, where visually analyzed by graphing contributions, importances against probability of a winning class.

Steps outlined above are few, in reality many process iterations had been taken, and more methods had been examined. More detailed description of a process along with justifications located in Appendix B.

4. ANALYSIS

4.1 Pre-processing and exploration.

In order to improve data convergence to true values, players, who played less than 100 hands were excluded from the analysis resulting in about 30k samples loss. (Achen 2013) There were 2 infinity and 5 NA values, observations were deleted. Finally, outliers had been handled by keeping data which is within 3 standard deviations of the mean. (209 samples loss) Poker hand data was reduced to 1 794 877 unique samples. (< 0.5% loss) Characteristics of 3073 players are contained in the final data frame, it was reduced from 4892. (about 35% loss). In the transition, from hand data frame to player, variables of interest were created (metadata in Appendix A):

1. **AF (aggression)**[11] = $\frac{Sum\ of\ Aggressive\ actions\ for\ all\ stages}{Sum\ of\ Passive\ actions\ for\ all\ stages}$.
2. **VPIP(loosiness**) = $\frac{Sum\ of\ times\ money\ were\ put\ voluntarily\ into\ pot}{Sum\ of\ hands\ played}$
3. **PFR(confidence**)=$\frac{Sum\ of\ times\ raised\ before\ flop}{Sum\ of\ hands\ played}$
4. **Decept(deceptiveness**)[12]=$\frac{Sum\ of\ Dceceptive\ actions\ for\ all\ stages}{Sum\ of\ stages\ played}$
5. **Commprop(opportunism**) = $\frac{Average\ amount\ of\ money\ put\ in\ the\ pot}{Average\ bankroll\ size}$
6. **Wtshwd(competitiveness**)=$\frac{Number\ of\ hands\ played\ until\ showdown}{Sum\ of\ hands\ played}$
7. **Profit_per_Vhand (profitability**) =

$$\frac{Average\ outcome\ at\ every\ hand}{Number\ of\ hands\ put\ money\ voluntarily}$$

Data was then explored using correlation (spearman) heatmap. Strong correlation between PFR and AF is obvious, this is expected because PFR is just subset of AF. Decept is also positively correlated with AF, which might indicate that players are more likely to fake weak, rather than strong play. Variables seem to have very low correlation with player profitability, except for COMMPROP, which express negative correlation. (figure 1)

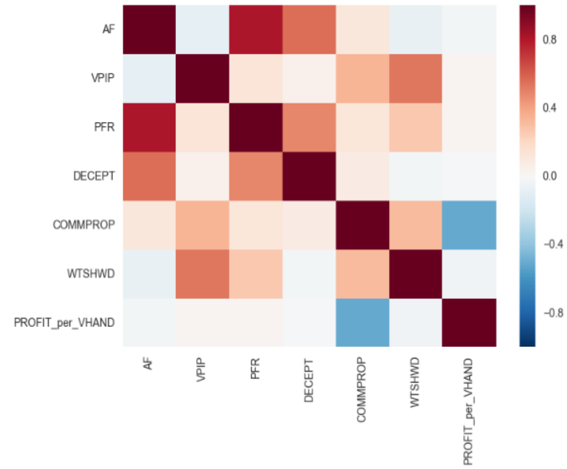

Figure 1. Correlation heatmap.

Distribution histograms and qq plots are examined, suggesting non-normality of data. (figure 2) Wilko-shapiro tests confirm it. (p<0.05)

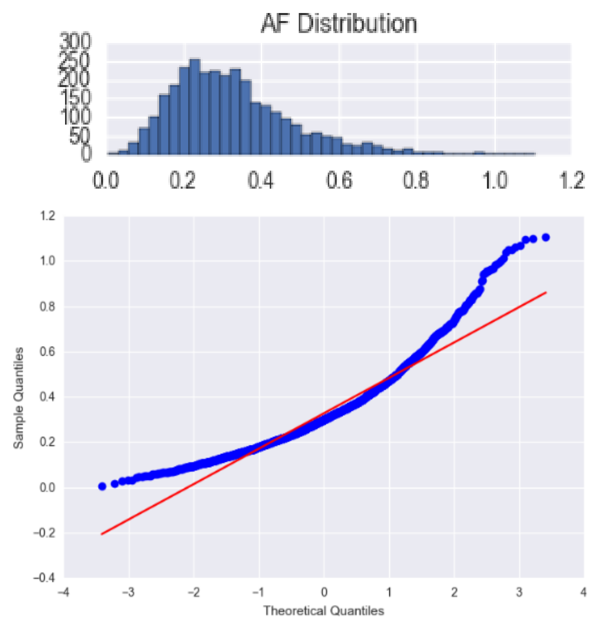

Figure 2. Distribution and QQ plots of AF.

4.2 Unsupervised clustering.

Three clustering models are investigated:

1. Borers' variables. AF, PFR, VPIP.

[8] OOB is the mean prediction error on each training sample x_i, using only the trees that did not have x_i in their bootstrap sample. (James et al. 2013)

[9] Description and derivation in http://blog.datadive.net/interpreting-random-forests/

[10] GINI impurity measures frequency a randomly chosen element would be incorrectly labeled if was randomly labeled according to distribution of labels in the sample.

[11] Aggressive action: bet/raise. Passive: checek/fold.

[12] Deceptive action: folding after betting; raising after checking, occurring one after another in same stage.

2. Expansion variables. Decept, Commprop, Wtshwd.
3. Full. AF, PFR, VPIP, Decept, Commprop, Wtshwd.

Number of clusters were chosen to equal 4 for first and second models, because it is the most common number, and leads to better comparison between two models and with literature. Silhouette plot on the left depicts aggregation of sample silhouette scores for each cluster referenced with average total score. Shapes which are uneven, below average and extending in the negative direction indicates poor clustering quality. It becomes apparent, that optimal number of clusters is between 3 and 5. Larger number leads to more overlapping and more uneven clusters; smaller number is not very justifiable for such diverse data. Elbow plot depicts *between sum of squares* [13]against number of clusters. It shows an edge for 4 clusters, as marginal increase in *between sum of squares* drops substantially at this level.

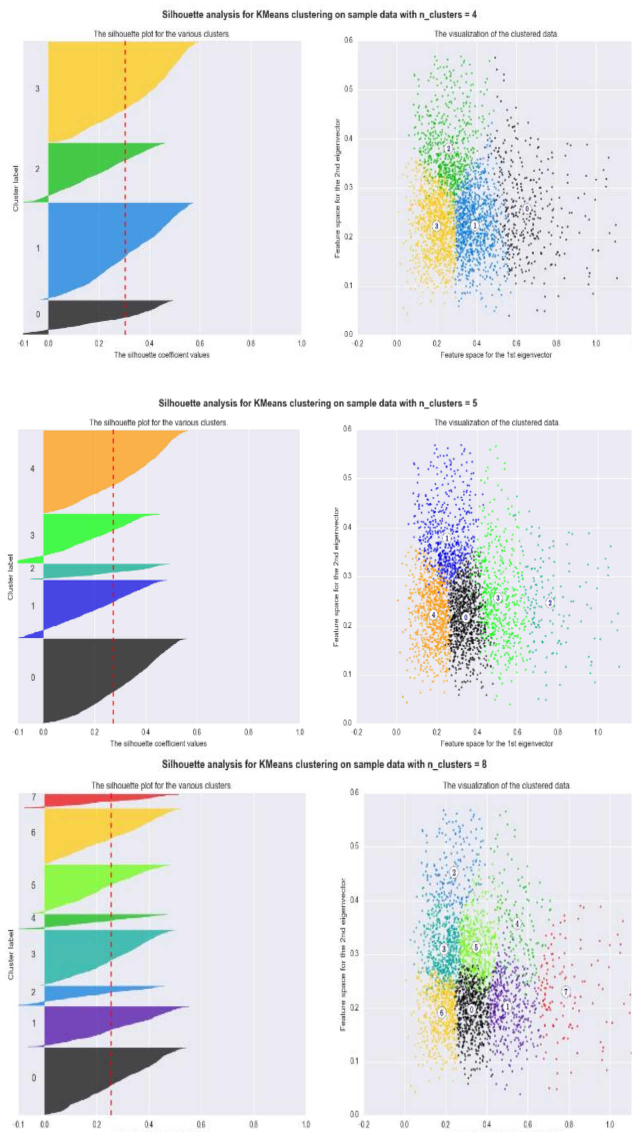

Figure 3. Silhouette score plots, and PCA reduced data points for Full model.

First two models expressed important difference, which becomes apparent using voronoi[14] diagrams. First model divides data very unevenly, probably as a result of higher density in regions. In figure 4, top picture, VPIP plotted against AF. Blue cluster could be seen as tight passive; brown – loose passive. Orange cluster is expected to be most profitable, as it is closest to what could be considered tight-aggressive[15]. Second model slices plane evenly and in straight lines. Clustering does not spot any blobs, and therefore is indifferent and divides data, rather than clusters. (figure 4) It is difficult to predict most profitable cluster.

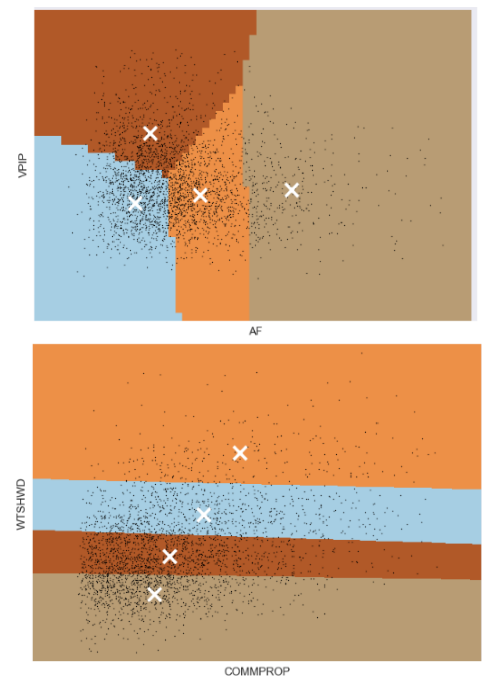

Figure 4. Voronoi diagrams. Top diagram is of first (Borers) model. Bottom diagram is of second (expansion) model.

Clustered data was exported to Tableau. PCMs separating clusters, are drawn. (figure 6) Proportion of winning players is included as last coordinate. Proportion of winning players in the whole dataset is referenced. Frequency distribution (percentage) histograms are drawn on the right. Bars are separated by the number of samples belonging to specific cluster. Cumulative distribution line added as second axis.

First model shows distinct differences in average characteristic values, for example red cluster has average AF level of 0.63 which is extremely aggressive, while blue one – 0.19, which is very passive. However, proportions of winning players are 0.31 and 0.3 accordingly. It is shockingly similar, given huge difference in aggression factor. Second model, however, shows a substantial difference in winners' proportion. Blue and red clusters are little bit above average profitability, (0.34, 0.33) while orange and green are below average. (0.15 and 0.23). Orange cluster shows highest COMMPROP and WTSHWD, suggesting that patience and calm, might save money for a poker player. Full model maintains similar characteristic differences, as first two models. It fails to show distinct differences in profitability. Distribution histograms and cumulative

[13] Between sum of squares = Total sum of squares – within sum of squares. Measures degree of heterogeneity of clusters.

[14] Voronoi diagram partitions plane into regions, so that every point in the region is closer to its center, than to the center of other regions.

[15] Tight-aggressive style is known to be most profitable. (North 2008)

line demonstrates negatively shifted distribution – 70% of players are losers.

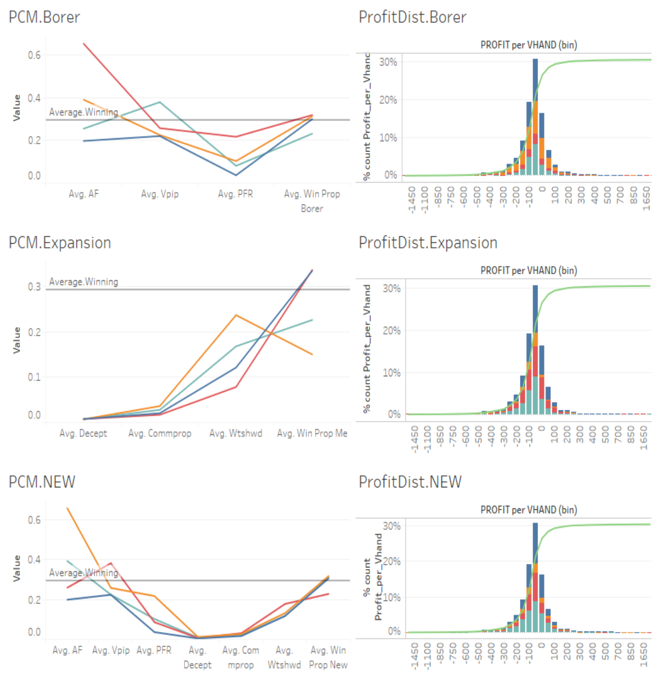

Figure 5. PCM along with profit distribution graphs for all three models.

To clarify the results, significance of differences need to be computed. Assumptions for one way ANOVA[16] are tested. QQ plots along with Wilko-Shapiro revealed non-normality early in exploration phase. Second assumption – homogeneity of variance is examined using *levene*[17] tests. Again, variances seem to be heterogeneous ($p < 0.05$). As a result, Welch t-tests were implemented instead of normal t-tests**.** However, it still requires approximate normality. Therefore, data was transformed using Box-Cox [18]lambda. Not completely normal, though transformed data is much more suitable for analysis. ANOVA and derived Wilko-Shapiro tests are robust to some violation of normality assumption.[19] Analysis confirms that differences in mean profitability are significant in second model, between clusters 1 and 2, 1 and 3, and 0 and 1. ($p<0.05$) There are no significant differences in other two models. PCM and Welch t-tests points that COMMPROP (opportunism) of over 3% and WTSHWD(competitiveness) over 10%, are associated with below average outcomes.

4.3 Supervised learning.

Failure to find profitable characteristics, despite quite distinct clustering left deep curiosity of what constitutes a winning player. In order to answer the question, winners (0< Profit_per_Vhand) were labeled as 1, loosers as 0. PCM of average characteristics is drawn, for two clusters. Two coordinate lines are surprisingly similar, which is discouraging. Frequency distribution plots for every characteristic are created that give a clearer picture. Additional line is added, describing expected distribution of characteristic for winners if there was absolutely no difference from that of losers.[20] Again, no distinct differences , except for COMMPROP. It shows that values of 0.25% to 0.75% are twice more frequent among winners. (figure 6) It is consistent with PCM in unsupervised clustering part, where second model showed distinct difference in profits between low and high COMMPROP clusters.

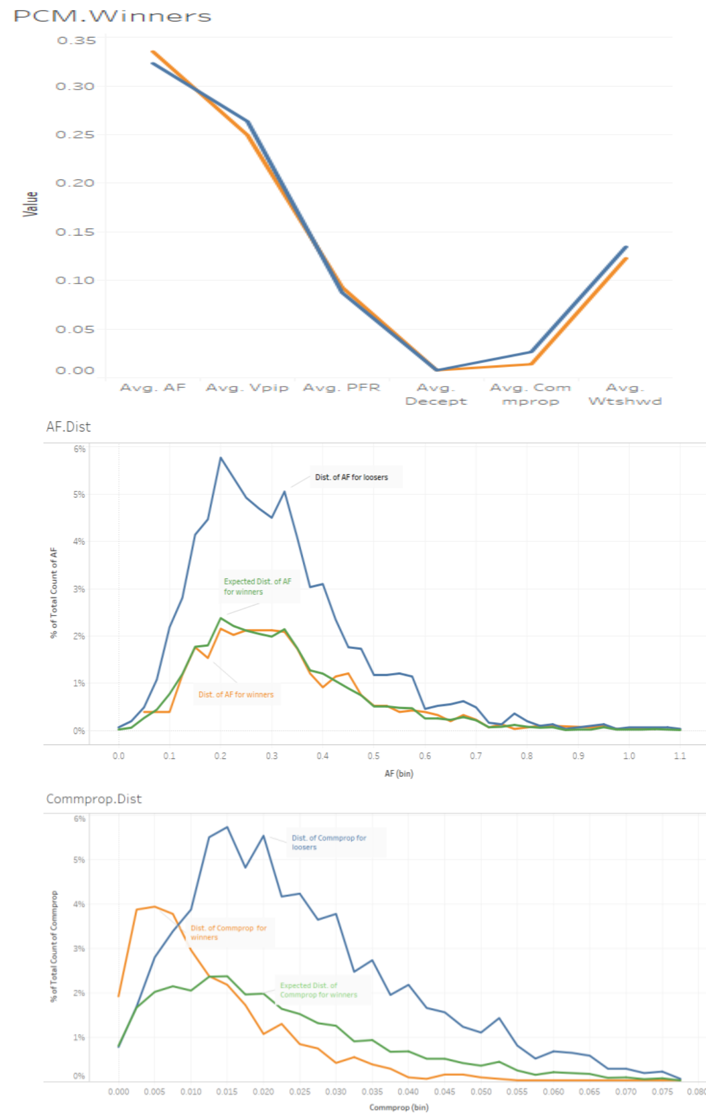

Figure 6. Top – PCM of characteristics. Middle – distribution of AF. Bottom – distribution of Commprop. Blue – winners, orange – losers, green – expected distribution for winners.

Random Forrest algorithm is implemented. Run with default parameters returns accuracy of nearly 100%, which is an obvious and significant overfitting. 10 fold cross validation accuracy is 74% which is much more realistic. Following steps are directed towards reducing the difference between cross validated and actual results. First, *Out-of-bag (OOB)* error rate is drawn against number of trees

[16] Analysis of variance. More: https://en.wikipedia.org/wiki/Analysis_of_variance

[17] More on *levene* test: https://en.wikipedia.org/wiki/Levene's_test

[18] More on Box-Cox: http://www.itl.nist.gov/div898/handbook/eda/section3/boxcoxno.htm

[19] More on robustness of t-tests: https://statistics.laerd.com/statistical-guides/one-way-anova-statistical-guide-3.php.

[20] Expected dist. Winners = ((Dist of winners and loosers) * 0.294). Because proportion of winners is 29.4% in a data.

to get an optimal number of decision tees. OOB error rate does not seem to drop, after 75 trees and retains levels of variance. (figure 8)

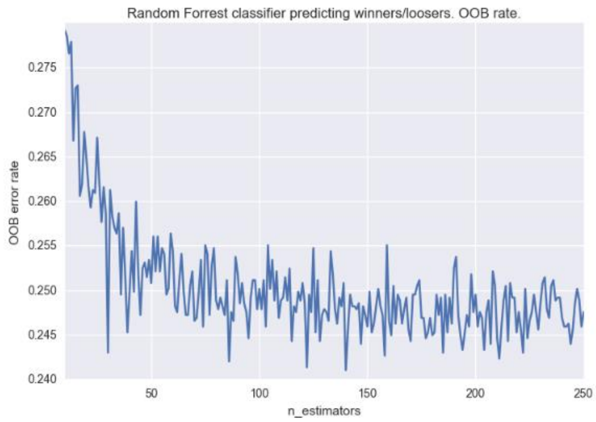

Figure 7. OOB rate against n_estimators for Random Forrest classifier.

Then several combinations of parameters are tested using GridSearchCv[21]. Best parameters[22] are chosen based on cross validated score. Gap between cross-validated score and actual score had been reduced further via experimentation with min_samples_leaf[23] parameter. (13 chosen instead of GridSearch suggested 3) The learning curves after fitting final model depicted in figure 8. Gap does not contract with additional training examples, however difference between cross-validated score and actual score is much more acceptable, if compared with initial model.

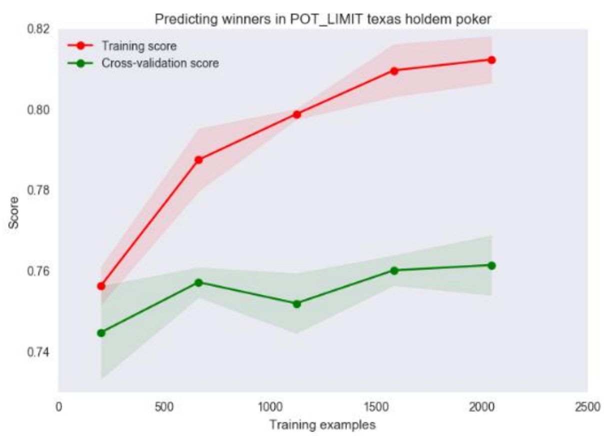

Figure 8. Learning curves for Random Forrest classifier.

Precision scores are similar for two clusters (0.82 and 0.8). However, there is substantial difference in recall – 0.95 for loosers; 0.48 for winners. Model achieves high accuracy score by predicting more frequent class. Classifier still performs much better than guessing - all loosers.(0.84 > 0.7) Next, feature importances are investigated. It is measured by decrease in GINI impurity and is implicit in the algorithm. Importances are depicted in histogram. (figure 9) COMMPROP is clearly much more important than remaining variables. Surprisingly, it is followed by control variable, measuring total number of hands played.

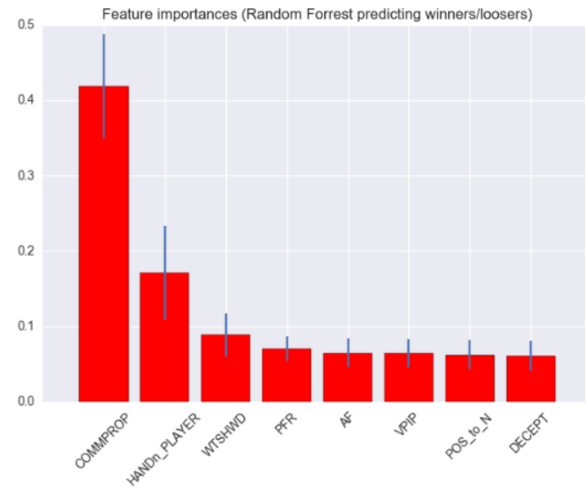

Figure 9. Feature importances, based on GINI impurity.

Python module *treeinterpreter* is utilized to get the feature contributions. It extracts the coefficients for different characteristics, which measures the numerical effect it had on probability for each decision. Results are exported back to Tableau for final visualizations. Feature contributions and actual values for features are depicted against probability of a winner class. Feature contributions are split on above/below average of actual value of a feature for that decision. The idea is to examine the non-linear possibility, that characteristic might have a greater (lesser) effect if is above (below) average value. Lines are added , which refer to frequency of winners in whole dataset (X axis). Reference regression lines added on Y axis. There are slight differences in contributions for above/below average characteristic split. For example, (figure 10) above average PFR is associated with greater positive contribution on winning probability; above average WTSHWD is associated with negative contributions. Larger the gap between regression lines, greater is the effect. Once more, COMMPROP plots are distinct as below average value is associated with greater contribution, and the effect is increasing with a decrease in COMMPROP.

[21] Method from sci-kit learn library. http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

[22] Best parameters: n_estimators=75; min_samples_leaf=5; min_samples_split=3; max_features = ‘sqrt’.

[23] Minimum required number of samples to split leaf node of decision tree

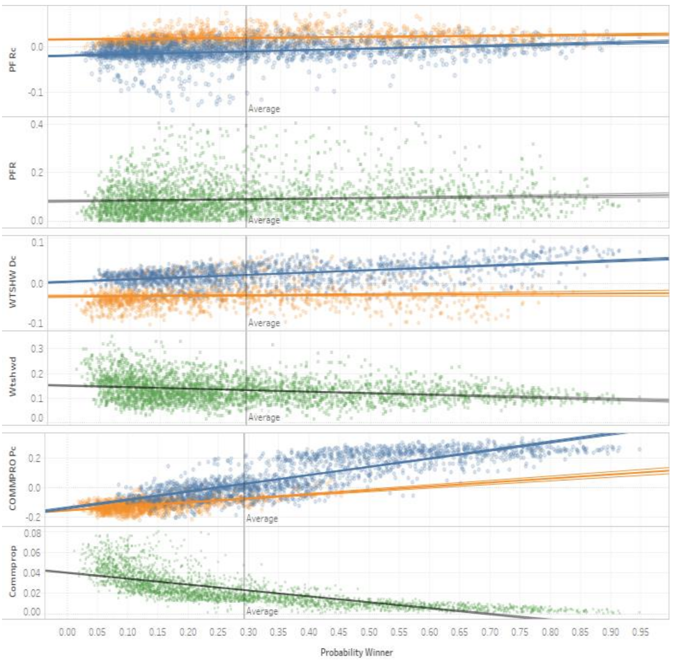

Figure 10. Feature contributions split by above/below average feature value. (orange/blue); actual feature values (green); both pictured against probability of a wining class.

5. Findings

First model (Borers') performed the worst, based on Silhouette and calinski-harabaz scores. (figure 11) Clusters had smaller average distance from each other, despite the fact that there were huge differences in some characteristics, such as *aggression*. (figure 5) Clustering in second model resulted in very different levels in *competitiveness* (wtshwd) and *profitability*. Worst cluster had 15% winning players, while best one – 33%. Worst cluster had highest levels of competitiveness (wtshwd) and opportunism (commprop). (0.23; 0.035) Third model performed average between remaining two, just with less distinction in profitability. Optimal number of clusters was found to be 4, using silhouette (figure 3) and elbow plots. To sum up, Borer's variables did not hold up as distinctly unique and significant, in player style clustering

	Silhoette score	Calinski-Harabaz	Winners proportion (best cluster)	Winners proportion(worst cluster)	Statistically significant mean differences
Borrers' variables(1st model)	0.31	2307.3	0.31	0.23	None
Expansion variables (2nd model)	0.38	4339.3	0.33	0.15	0 and 1, 0 and 3, 1 and 2 clusters.
Full (3rd model)	0.34	2507.7	0.31	0.22	None

Figure 11. Results of unsupervised clustering.

Attempt to uncover characteristics of a winning poker player was marginally successful. Distribution graphs of winners, revealed unexpectedly large frequencies for small magnitudes of COMMPROP. Random Forest was able to predict winners/losers class with 75% cross-validated accuracy. However, it was much more successful in recalling losers than winners (0.95 and 0.48). GINI importance measure revealed COMMPROP significance, at value of 0.41, while second most important measure Handn_Players stood at 0.17. Contributions were close to 0, meaning that variables influenced probability non-linearly. Positive effects of certain values, were canceled out by negative effects of other ones; or the effects were non-deterministic. Graphing feature contributions, next to actual values and splitting by bellow/above average, revealed certain marginal differences. (figure 10) It also firmly confirmed, the negative relationship between opportunism (COMMPROP) and profitability.

6. Critical reflection

There were two main functions of visual analytics in this project. First, it helped to justify and evaluate. For example: qq plots (figure 2) Silhoette plots (figure 3); OOB error rate. (figure 7). Second function – summarize the results. For instance: PCMs (figure 5), histograms (figure 9), graphs (figure 10). Analogy would be, if engine of a vehicle is machine learning algorithms and statistical models; roads and paths are data, then body, windows and steering wheel are visual analytics. Destination here, was to cluster poker player styles, and deduce characteristic values which increases likelihood of winning. Reader was supposed to be taken, on a road trip in vehicle whose engine is powered by K-Mans clustering, and RandomForrest algorithms. View from a window and steering actions, were defined by colorful, intuitive and effective visual techniques. Results are satisfying, combining Python and Tableau proved really powerful. For justification, and evaluation tasks, python visualization methods seemed more convenient, while Tableau was irreplaceable for visual summary task.

Poker variation analyzed here was POT LIMIT Hold'em, it lies in between LIMIT Hold'em and NO-LIMIT Hold'em, on maximum amount of bet allowed. As of today, POT LIMIT Hold'em is the least popular out of three. Most studies and papers specialize on remaining two variations. Borer (2007) based on LIMIT Hold'em and Harrington (2004) based on NO-LIMIT Hold'em. However, only difference among variations is betting structure. It is likely to result in different absolute characteristic values, however comparative relationship and significance expected to hold. For example, low COMMPROP is expected to be associated with higher winning percentage in NO-LIMIT Hold'em. However, it does not necessarily need to be below 2% bankroll, it threshold could be higher or lower. (figure 5, middle) Nonetheless, it is possible, that Borers' variables could lead to significantly better clustering (based on Silhouette and Calinski-Harabaz) in NO-LIMIT Hold'em or LIMIT Hold'em. However, poker variations are so similar, that it is highly unlikely.

Methods used in this paper are quite universal. Both machine learning methods, k-means clustering and random forests are used anywhere from tumor detection (Mandwe et.al 2016) to stock market analysis. (Khaidem et.al 2016) Visualization techniques for justification and evaluation are general to particular method. (OOB error plot for Random Forest) PCMs of averages are useful, when dealing with high-dimensional data, where means are relevant. It is great for visualization of clustering results, as differences among clusters could be observed in one graph. If averages are not relevant, quality of PCMs' quickly diminishes with increase in data, as overclutting occurs. It could be managed through transparency and color, however small multiples of separate coordinate graphs might be more convenient. Graphs are most popular for time-series visualization, due to continuous nature of this dimension, and effectiveness of positioning and angle channels. (Munzner 2014) Probabilities, returned by Random Forest are continuous, therefore graph was chosen. Picturing contributions and actual values, against the probability measure is effectively visualizes explanatory performance of an algorithm. (figure 10)

There are some important caveats that need to be outlined. First, deceptiveness measure (decept) universally was very infrequent (0.7%), which resulted in many players having 0 number of deceptive stages. Also, the mean differences of DECEPT for various clusters was not as substantial, as differences of other characteristics. Second caveat, is age of data. It was generated in late 1990s, before online poker boom. It could be argued, that game had evolved since then. Last important limitation, is that in the absence of *hand strenght*[24] variable, it is tricky to deduce unique player style. Danger for spurious variables arise, COMMPROP could be one of them, as significant effect discovered, could be barely a result of winning players having larger bankroll, and therefore not needing to commit higher bankroll percentage. Analysis, could be replicated using more recent data, not limited to one poker variation, and expanded on number of variables investigated.

7. BIBLIOGRAPHY

1. Gareth James; Daniela Witten; Trevor Hastie; Robert Tibshirani , *(2013). An Introduction to Statistical Learning. Springer. pp. 316–321.*
2. Saabas, Ando. "Random Forest Interpretation with Scikit-learn." *Diving into Data*. N.p., 12 Aug. 2015. Web. 13 Dec. 2016.
3. Tamara Munzner. (2014) ‘’*Analysis and Design*.’’ A.K. Peters' Visualization Series. CRC Press.
4. Achen, A.L. (2013) ‘Enhancing Poker Agents with Hand History Statistics’, .
5. Bowling, M., Burch, N., Johanson, M. and Tammelin, O. (2015) ‘Heads-up limit hold’em poker is solved’, *Science*, 347(6218), pp. 145–149. doi: 10.1126/science.1259433.
6. Caruana, R. and Niculescu-Mizil, A. (2006) ‘An Empirical Comparison of Supervised Learning Algorithms’, .
7. Harrington, D. and Robertie, B. (2004) *Harrington on hold’em: Expert strategy for no-limit tournaments: Volume 1: Strategic play*. Henderson, NV: Two Plus Two Publishing.
8. Moser, B.K. and Stevens, G.R. (1992) ‘Homogeneity of variance in the Two-Sample means test’, *The American Statistician*, 46(1), p. 19. doi: 10.2307/2684403.
9. North, G. (2009) ‘.Development of a poker hand analysis and table persona classifier.’
10. Hellmuth, P. (2003). *Play Poker Like the Pros.* New York: HarperCollins.
11. Borer, T., Mak, L., & Tanenbaum, B. (2007). ‘Limit Hold'em: Winning shorthanded strategies.’ Hassocks: D & B Publishing.
12. Ashwini A.Mandwe, Aisa Anjum. ‘’Detection of brain tumor using k-means clustering’’ (2016) International Journal of Science and Research (IJSR), 5(6), pp. 420–423. doi: 10.21275/v5i6.nov164061.
13. Khaidem, L., Saha, S. and Dey, S. (2016) ‘Predicting the direction of stock market prices using random forest’.

[24] Hand-Strenght could be deduced by the percentage of hands it beats given random community cards.

APPENDIX A

INDEX	NAME_ID	Unique id for every player.	Unique string.
VARIABLES OF INTERST (PLAYERS' SUCCESS)	PROFIT_per_VHAND	Amount of money won per hand played.	Float[-inf, inf]
	TOTAL_PROFIT	Total amount of return money won. (negative if money lost)	Integer[-inf, inf]
COUNT VARIABLES (USED FOR TRANSFORMATIONS)	AGRO_N	Number of stages aggressive action performed by a player.	Integer
	PASS_N	Number of stages passive action performed by a player.	Integer
	DECEPT_N	Number of stages deceptive action performed by a player.	Integer
	STAGES_N	Number of stages played by a player.	Integer
	HANDn_PLAYERS	Number of hand splayed by a player.	Integer
Borer's *holy trinity*	AF	Proportion of aggressive stages relative to passive.	Proportion [0,1]
	VPIP	Proportion of hands, player put money into the pot voluntaril	Proportion [0,1]
	PFR	Proportion of hands, player raised in pre-flop stage, relative t	Proportion [0,1]
Expansion variables	DECEPT	Proportion of deceptive stages, relative to total number of st	Proportion [0,1]
	COMMPROP	Mean proportion of stack played in a hand.	Proportion [0,1]
	WSHWD	Proportion of hands player participated until showdown stag	Proportion [0,1]
Control variables (position)	POS_mean	Average of players position at a table.	Float
	PLAYERSn_mean	Average of a number of players participated in a hand.	Float
	POS_to_N	Position, relative to total number of players.	Proportion [0,1]

Appendix A. Description of variables.

APPENDIX B

Analytical approach	Software/library	Type	Justification
Rewrite seperate files into one csv file one after another. Check for errors and set data types. Read into a data frame.	SQL. Python. (pandas)	C	Necessary step.
Delete data of players with <100 hands played.	Python.	C	**(Achen 2013**) shows good convergence at 100-200 hand range.
Derive variables using computational transformations. (more in Appendix A)	Python.	C	Borer's **(Borer 2007)** ''holy trinity.'' Authors personal opinion.
Case deletion.	Python.	C	Values are missing at random and represent less than 5% of data.
Correlation heatmap; distribution histograms; QQ plots.	Python.	VA	Get the sense of data interaction and distribution properties. Check normality.
Unsupervised K-means (k=4) clustering.	Python. (scikit; numpy)	C	K-means has great scalability, speed, fits better with few clusters. (http://scikit-learn.org/stable/modules/clustering.html)
Silhouette coefficient; calinski-harabaz score; PCA(n=2);voronoi diagrams; 3d cube	Python. (scikit; numpy)	C/VA	Two scores are main measures for evaluating unsupervised learning algorithm, as pointed in scikit-learn.org. Voronoi diagrams (after PCA) and 3d cube help to understand how players are seperated. (uniformly/patterns)
Unsupervised K-means (k=4) clustering.	Python. (scikit; numpy)	C	*See above.*
Silhouette coefficient; calinski-harabaz score; PCA(n=2); voronoi diagrams; 3d cube	Python. (scikit; numpy)	C/VA	*See above.*
Unsupervised K-means (k=3,4,5,8,10) clustering.	Python. (scikit; numpy, matplotlib)	C	*See above.*
Silhouette coefficient; calinski-harabaz score; PCA(n=2); voronoi diagrams; 3d cube; silhouette plot.	Python. (scikit; numpy, matplotlib)	C/VA	Silhoette plots separate shapes for different clusters, volume proportional to number of samples in a cluster. Average line, helps to deduce underperforming clusters. More even plot, with less values bellow 0 is preffered. (http://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html#sphx-glr-auto-examples-cluster-plot-kmeans-silhouette-analysis-py)
Parallel coordinates map.	Tableau.	VA	Parallel coordinates is efficient way to depict multidimensional data, as it primary utlizes location, angle, coloring, which are highly expressive cues.**(Munzner, 2014)**Averages are computed to avoid clutter. Extra coordinate added, to depict proportion of winning players for each cluster.
Distribution plot (histogram). Analysis of variance. (Welch t-tests)	Tableau. Python. (scikit-learn; stats; numpy)	C/VA	Histogram is ''go to'' method visualize distributions, and distribution is perfect for understanding style profitability. Welch t-test is robust to unequal sample variances, and is superior to t-tests. (Moser and Stevens 1992)
Parallel coordinates map.	Tableau.	VA	Shows averages for all characteristics, for all clusters in one simple plot.
Random Forrest algorithm.	Python.(scikit-learn,pandas , numpy)	C	Random forrest is one of the most robust and accurate algorithms. **(Breinman 2001)** It does not require assumptions, such as normality which is violated in this project.
OOB plot, learning curve; confusion matrix; ROC-AUC curves.	Python. (scikit-learn, pandas, numpy)	C/VA	Out of bag error plot helps to find optimal number of trees; learning curve shows how useful is the sample size for the algorithm; confusion matrix and ROC-AUC curves depicts the performance and accuracy.
Multiple graphs.	Tableau. Python. (tree interpreter)	VA	Graph is the most common method for time series visualization, however continous probability instead of time is used on the axis and feautures' contribution to decision on y axis. Shows how important was the feature for defining winner. Treeinterpreter was used to deduce contributions. **(Saabas 2014)**

Appendix B. Analytical steps. Abbreviations in *type* column: C for computation; VA for visual analysis technique.