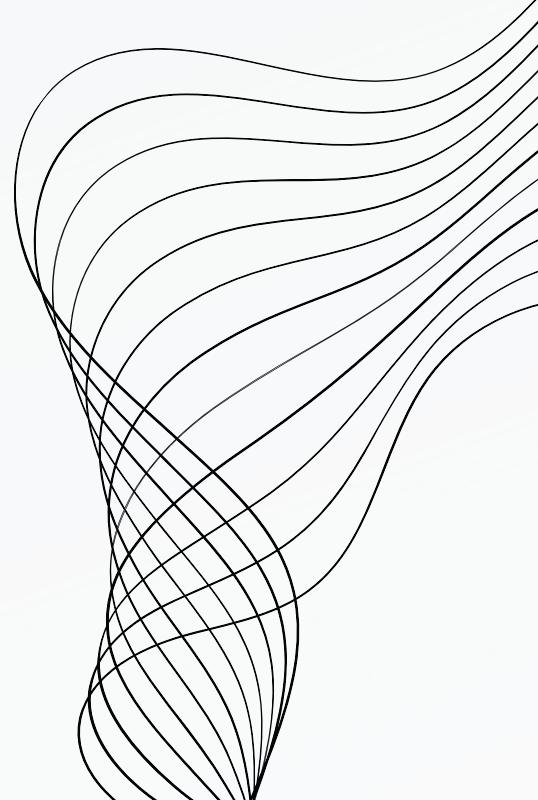


# MIND GAME INSIGHTS

A DEEP DIVE INTO GAMING AND PSYCHOLOGICAL WELL-BEING RELATIONSHIPS

COURSE: INST 737 - INTRODUCTION TO DATA SCIENCE



MILESTONE 3

TEAM:

**RAJEEVAN MADABUSHI  
PRANAV ADIRAJU  
ASMITA SAMANTA**

# RESEARCH QUESTION

- **Dependent Variable**

*Generalized Anxiety Disorder (GAD\_T)*

- **Independent Variables**

*Narcissism, Social Phobia Inventory (SPIN\_T),  
Hours, Satisfaction With Life (SWL\_T), Reasons to  
Play (whyplay\_clean), Status of Work (Work),  
Style of Play (Playstyle\_clean)*

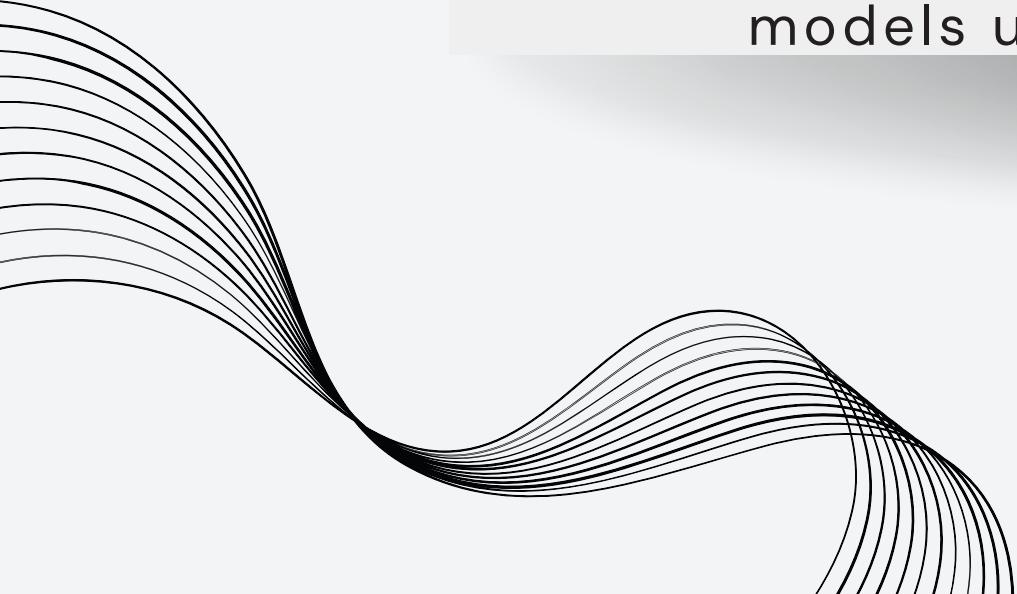
# THE RECAP

## MILESTONE 1

- In Milestone 1, we conducted preliminary data cleaning and performed Exploratory Data Analysis of our dataset.

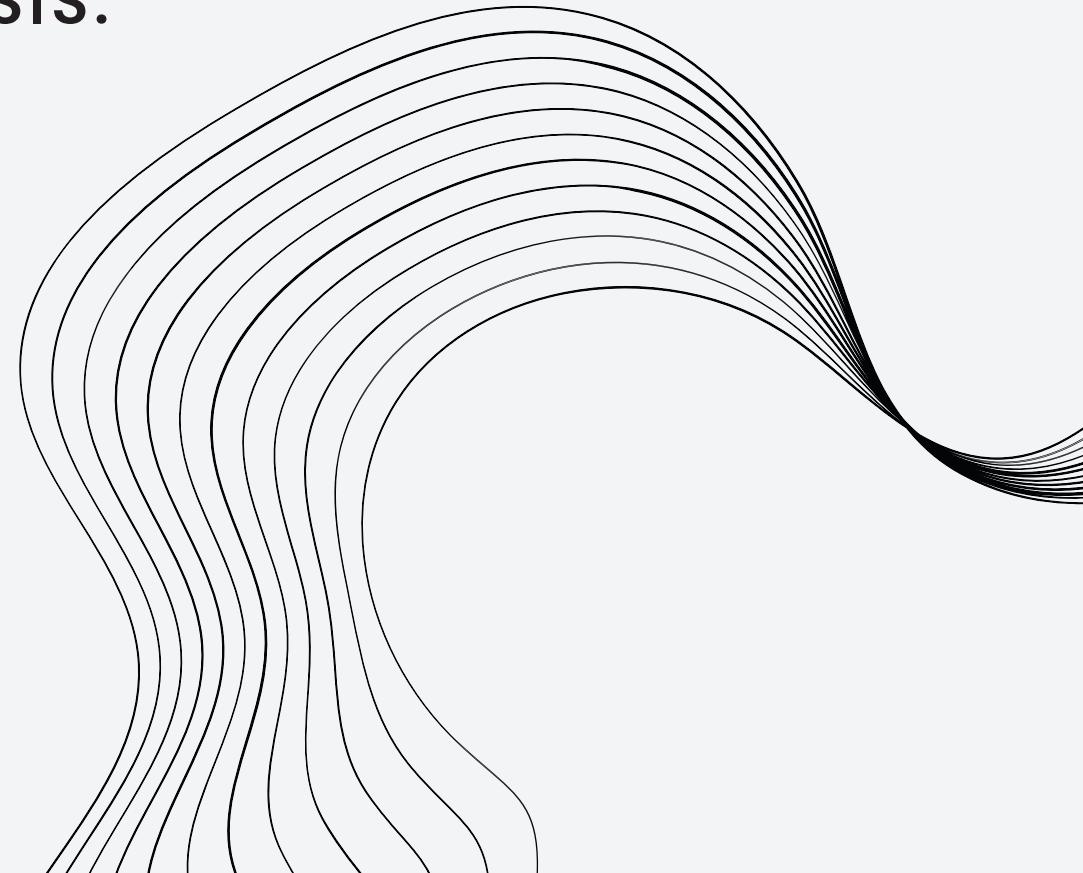
## MILESTONE 2

- In Milestone2, We explored various regression and classification models, experimented with Decision Trees with ensemble methods like Bagging and Random Forests. and ending up with a comparative analysis of all the models used.



# MILESTONE 3

FOR MILESTONE 3, WE TRAINED AND TESTED VARIOUS SVM MODELS, EXPLORED THE APPLICABILITY OF NEURAL NETWORKS FOR OUR RESEARCH QUESTION AND INCORPORATED VARIOUS CLUSTERING TECHNIQUES AND REPORTED RELEVANT STATISTICS. FURTHERMORE, WE PERFORMED A COMPARATIVE ANALYSIS AND APPLIED FEATURE SELECTION TO FEW OF MODELS FROM MILESTONE 2 AND MILESTONE 3. FINALLY, WE HAVE ALSO EXPLORED THE POTENTIAL ETHICAL IMPLICATIONS OF OUR RESEARCH HYPOTHESIS.



# SVM (STATE VECTOR MACHINES)

## OUR APPROACH

FOR OUR HYPOTHESIS, WE TREATED ANXIETY LEVELS (GAD\_T) AS A CATEGORICAL VARIABLE, USING A ONE-VS-ONE SVM CLASSIFICATION MODEL RATHER THAN A NUMERICAL VARIABLE AND TREATING IT AS A REGRESSION PROBLEM.

## DATA PRE-PROCESSING

WE THEN IMPORTED THE DATASET, SELECTED RELEVANT COLUMNS, CONVERTED GAD\_T INTO A FACTOR, AND SPLIT THE DATA INTO AN 80-20 TRAINING AND TESTING SET FOR OUR MODEL PREDICTION.



# SVM (STATE VECTOR MACHINES)

## MODEL TRAINING & EVALUATION

HERE ARE THE SUMMARY OF KERNELS THAT WERE USED FOR OUR RESEARCH HYPOTHESIS:

Kernel Type	Support Vectors	Training Error	Overall Accuracy
Gaussian	8420	79.22%	14.35%
Polynomial	8372	84.39%	12.56%
Euclidean	8372	84.42%	12.65%
Hyperbolic Tangent	8310	89.22%	11.90%

NOTE: ALTHOUGH, ALL THE KERNELS DID SHOW A LIMITED PERFORMANCE FOR OUR DATASET, GAUSSIAN KERNEL ('RBFDOT') WAS THE BEST AMONG THE LOT.

# SVM (STATE VECTOR MACHINES)

## CLASS-WISE STATISTICS OF EACH KERNEL

### 1.GAUSSIAN KERNEL

	Statistics by Class:											
	Class: 0	Class: 1	Class: 2	Class: 3	Class: 4	Class: 5	Class: 6	Class: 7	Class: 8	Class: 9	Class: 10	Class: 11
Sensitivity	0.52823	0.13744	0.20147	0.19383	0.081218	0.037500	0.066176	0.032787	0.027778	0.00000	0.000000	0.0200000
Specificity	0.76043	0.88568	0.81355	0.78265	0.941697	0.968335	0.955096	0.982966	0.985572	1.00000	0.994645	0.9990329
Pos Pred Value	0.22625	0.11741	0.13784	0.09670	0.125000	0.088235	0.091837	0.105263	0.093750	Nan	0.00000	0.3333333
Neg Pred Value	0.92398	0.90273	0.87318	0.88996	0.909045	0.924878	0.937129	0.943269	0.949664	0.96364	0.969625	0.9768322
Prevalence	0.11709	0.09962	0.12890	0.10718	0.093012	0.075543	0.064212	0.057602	0.050992	0.03636	0.030217	0.0236072
Detection Rate	0.06185	0.01369	0.02597	0.02077	0.007554	0.002833	0.004249	0.001889	0.001416	0.00000	0.000000	0.0004721
Detection Prevalence	0.27337	0.11662	0.18839	0.21483	0.060434	0.032106	0.046270	0.017941	0.015109	0.00000	0.005194	0.0014164
Balanced Accuracy	0.64433	0.51156	0.50751	0.48824	0.511458	0.502918	0.510636	0.507876	0.506675	0.50000	0.497322	0.5095164
	Class: 12	Class: 13	Class: 14	Class: 15	Class: 16	Class: 17	Class: 18	Class: 19	Class: 20	Class: 21		
Sensitivity	0.0476190	0.065217	0.0000000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.0500000		
Specificity	0.9913295	0.985521	0.9995208	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.9976168		
Pos Pred Value	0.1000000	0.090909	0.0000000	Nan	Nan	Nan	Nan	Nan	Nan	0.1666667		
Neg Pred Value	0.9809342	0.979376	0.9853566	0.98536	0.98725	0.992918	0.99339	0.992918	0.998111	0.9910038		
Prevalence	0.0198300	0.021719	0.0146364	0.01464	0.01275	0.007082	0.00661	0.007082	0.001889	0.0094429		
Detection Rate	0.0009443	0.001416	0.0000000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.0004721		
Detection Prevalence	0.0094429	0.015581	0.0004721	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.0028329		
Balanced Accuracy	0.5194743	0.525369	0.4997604	0.50000	0.50000	0.50000	0.50000	0.50000	0.50000	0.5238084		

# SVM (STATE VECTOR MACHINES)

## CLASS-WISE STATISTICS OF EACH KERNEL

## 2. POLYNOMIAL KERNEL

	class: 0	class: 1	class: 2	class: 3	class: 4	class: 5	class: 6	class: 7	class: 8	class: 9	class: 10	class: 11
Sensitivity	0.5363	0.10900	0.30769	0.10132	0.0101523	0.0000000	0.00000	0.00000	0.00000	0.00000	0.0156250	0.00000
Specificity	0.7283	0.90299	0.59295	0.82866	0.9588756	0.9994893	1.00000	1.00000	1.00000	1.00000	0.9985394	1.00000
Pos Pred Value	0.2075	0.11058	0.10060	0.06628	0.0246914	0.0000000	NaN	NaN	NaN	NaN	0.2500000	NaN
Neg Pred Value	0.9221	0.90157	0.85269	0.88481	0.9042710	0.9244214	0.93579	0.9424	0.94901	0.96364	0.9701957	0.97639
Prevalence	0.1171	0.09962	0.12890	0.10718	0.0930123	0.0755430	0.06421	0.0576	0.05099	0.03636	0.0302172	0.02361
Detection Rate	0.0628	0.01086	0.03966	0.01066	0.0009443	0.0000000	0.00000	0.00000	0.00000	0.00000	0.0004721	0.00000
Detection Prevalence	0.3026	0.09821	0.39424	0.16383	0.0382436	0.0004721	0.00000	0.00000	0.00000	0.00000	0.0018856	0.00000
Balanced Accuracy	0.6323	0.50600	0.45032	0.46499	0.4845139	0.4997446	0.50000	0.50000	0.50000	0.50000	0.5070822	0.50000
sensitivity	0.00000	0.0000000	0.00000	0.00000	0.00000	0.0000000	0.00000	0.00000	0.00000	0.00000	0.0000000	0.00000
specificity	1.00000	0.9995174	1.00000	1.00000	1.00000	1.0000000	1.00000	1.00000	1.00000	1.00000	1.0000000	1.00000
Pos Pred Value	NaN	0.0000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Neg Pred Value	0.98017	0.9782711	0.98536	0.98536	0.98725	0.992918	0.99339	0.992918	0.998111	0.990557		
Prevalence	0.01983	0.0217186	0.01464	0.01464	0.01275	0.007082	0.00661	0.007082	0.001889	0.009443		
Detection Rate	0.00000	0.0000000	0.00000	0.00000	0.00000	0.0000000	0.00000	0.00000	0.00000	0.00000	0.0000000	0.00000
Detection Prevalence	0.00000	0.0004721	0.00000	0.00000	0.00000	0.0000000	0.00000	0.00000	0.00000	0.00000	0.0000000	0.00000
Balanced Accuracy	0.50000	0.4997587	0.50000	0.50000	0.50000	0.5000000	0.50000	0.50000	0.50000	0.50000	0.5000000	0.50000

# SVM (STATE VECTOR MACHINES)

## CLASS-WISE STATISTICS OF EACH KERNEL

### 3. EUCLIDEAN KERNEL

# SVM (STATE VECTOR MACHINES)

## CLASS-WISE STATISTICS OF EACH KERNEL

### 4.HYPERBOLIC TANGENT KERNEL

Statistics by Class:											
Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6 Class: 7 Class: 8 Class: 9 Class: 10 Class: 11											
Sensitivity	0.4637	0.11374	0.17582	0.10573	0.020305	0.0125000	0.000000	0.131148	0.083333	0.0259740	0.109375
Specificity	0.7171	0.91557	0.85962	0.90534	0.966163	0.9897855	0.995964	0.830160	0.951244	0.9857913	0.940117
Pos Pred Value	0.1786	0.12973	0.15635	0.11823	0.057971	0.0909091	0.000000	0.045070	0.084112	0.0645161	0.053846
Neg Pred Value	0.9098	0.90326	0.87576	0.89399	0.905808	0.9246183	0.935545	0.939875	0.950771	0.9640632	0.971328
Prevalence	0.1171	0.09962	0.12890	0.10718	0.093012	0.0755430	0.064212	0.057602	0.050992	0.0363551	0.030217
Detection Rate	0.0543	0.01133	0.02266	0.01133	0.001889	0.0009443	0.000000	0.007554	0.004249	0.0009443	0.003305
Detection Prevalence	0.3041	0.08735	0.14495	0.09585	0.032578	0.0103872	0.003777	0.167611	0.050519	0.0146364	0.061379
Balanced Accuracy	0.5904	0.51466	0.51772	0.50553	0.493234	0.5011427	0.497982	0.480654	0.517289	0.5058827	0.524746
Class: 12 Class: 13 Class: 14 Class: 15 Class: 16 Class: 17 Class: 18 Class: 19 Class: 20 Class: 21											
Sensitivity	0.00000	0.000000	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.0500000	
Specificity	0.99326	0.994208	0.998563	0.996646	1.00000	0.992392	1.00000	1.000000	1.000000	0.9985701	
Pos Pred Value	0.00000	0.000000	0.000000	0.000000	NaN	0.000000	NaN	NaN	NaN	0.2500000	
Neg Pred Value	0.98004	0.978158	0.985343	0.985315	0.98725	0.992864	0.99339	0.992918	0.998111	0.9910123	
Prevalence	0.01983	0.021719	0.014636	0.014636	0.01275	0.007082	0.00661	0.007082	0.001889	0.0094429	
Detection Rate	0.00000	0.000000	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.0004721	
Detection Prevalence	0.00661	0.005666	0.001416	0.003305	0.00000	0.007554	0.00000	0.000000	0.000000	0.0018886	
Balanced Accuracy	0.49663	0.497104	0.499281	0.498323	0.50000	0.496196	0.50000	0.500000	0.500000	0.5242850	

# CLUSTERING

## DATA PREPROCESSING

- *Environment setup to run R in a Python workspace*
- *Installation of 13 packages across data transformation, visualization and ML model building*
- *Loading dataset with required columns*

## MODEL TRAINING & EVALUATION

- *Three selected clustering methods – PAM, Hierarchical and DBSCAN*
- *Usage of Gower distance for model building*

## MODEL

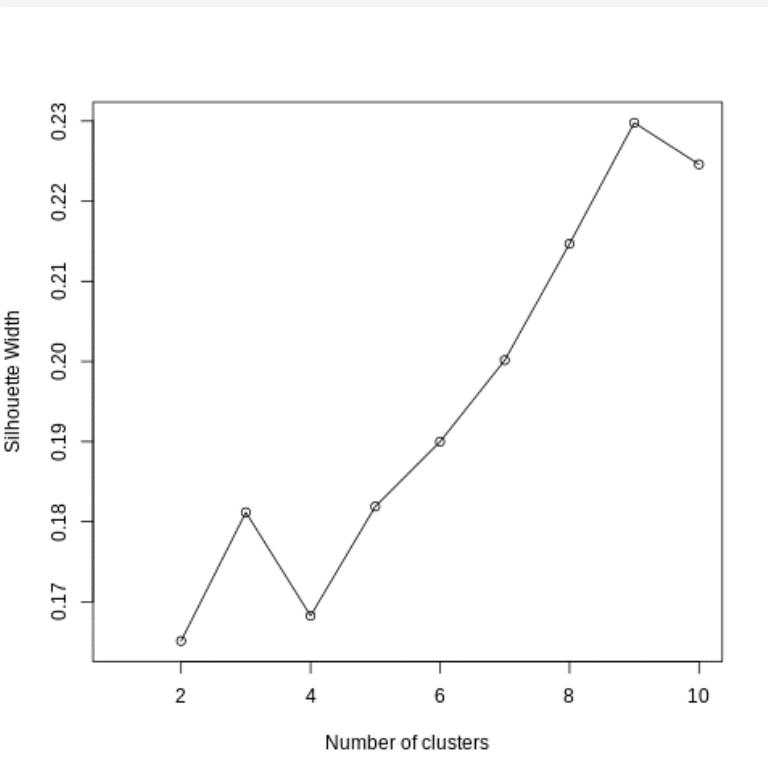
- *Below components are calculated for each clustering method –*
  - a. *Optimal no. of clusters*
  - b. *Elements per cluster*
  - c. *Structural plots*
  - d. *Interpretation of Clusters*



# CLUSTERING

## PAM CLUSTERING

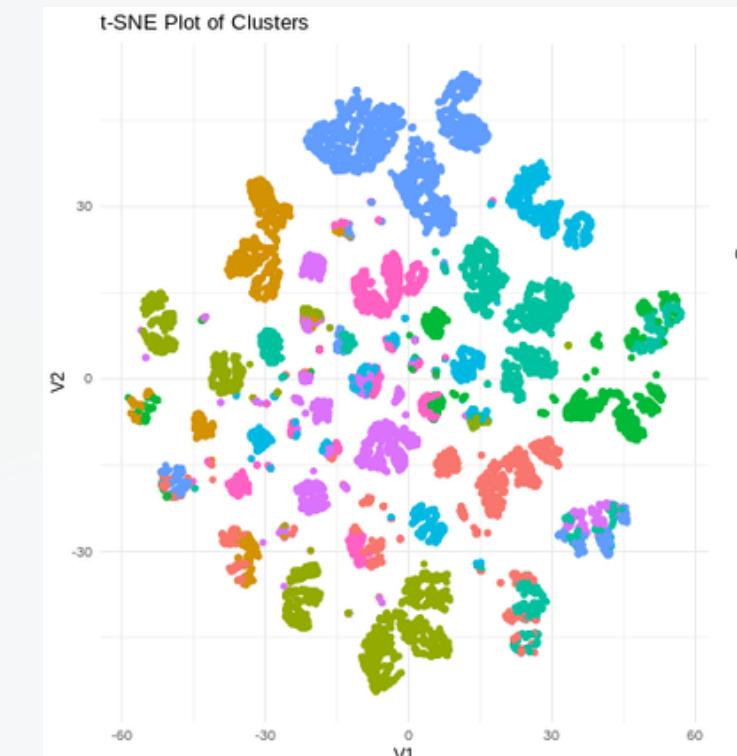
Optimal No. of Clusters



Elements per cluster

Cluster	Elements
1	1085
2	851
3	1501
4	888
5	1554
6	1093
7	1679
8	1084
9	856

Structural Plot



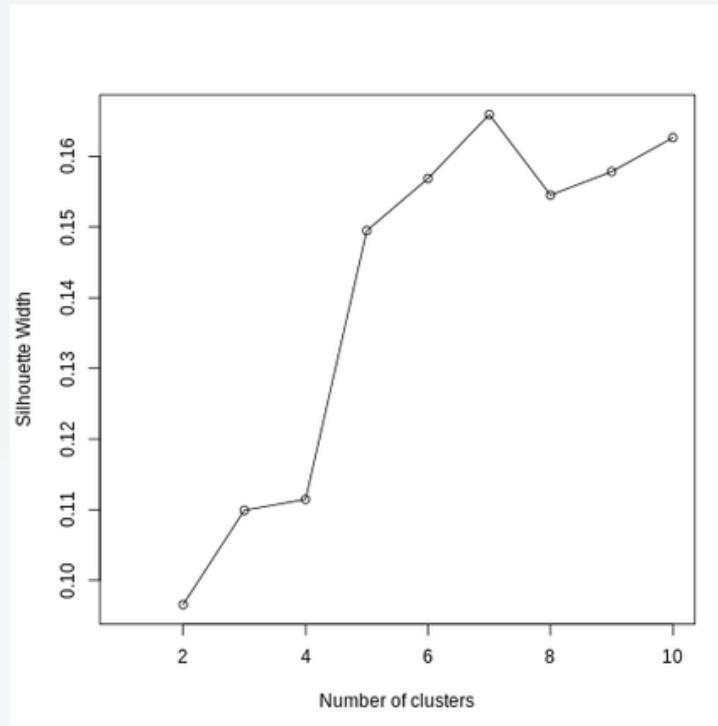
Interpretation of clusters

Cluster	GAD_T (Min-Max)	Narcissism (Min-Max)	SPIN_T (Min-Max)	Hours (Min-Max)	SWL_T (Min-Max)	Top whyplay_clean	Top Work Category	Top Playstyle_clean
1	0.0 - 21.0	1.0 - 5.0	0.0 - 62.0	0.0 - 100.0	5.0 - 35.0	fun (724)	Employed (1039)	Online w/ RL friends (776)
2	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	0.0 - 92.0	5.0 - 35.0	fun (813)	Students (573)	Online w/ strangers (828)
3	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	1.0 - 120.0	5.0 - 35.0	improving (1434)	Students (775)	Online w/ strangers (1454)
4	0.0 - 21.0	1.0 - 5.0	0.0 - 62.0	1.0 - 80.0	5.0 - 35.0	fun (563)	Students (871)	Online w/ RL friends (706)
5	0.0 - 21.0	1.0 - 5.0	0.0 - 64.0	2.0 - 105.0	5.0 - 35.0	improving (1394)	Students (1076)	Online w/ RL friends (1532)
6	0.0 - 21.0	1.0 - 5.0	0.0 - 67.0	2.0 - 120.0	5.0 - 35.0	improving (941)	Students (577)	Online w/ acquaintances (1021)
7	0.0 - 21.0	1.0 - 5.0	0.0 - 63.0	0.0 - 85.0	5.0 - 35.0	fun (1495)	Students (1583)	Online w/ RL friends (1639)
8	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	0.0 - 120.0	5.0 - 35.0	winning (1006)	Students (579)	Online w/ strangers (761)
9	0.0 - 21.0	1.0 - 5.0	0.0 - 67.0	1.0 - 100.0	5.0 - 35.0	fun (698)	Students (526)	Online w/ acquaintances (797)

# CLUSTERING

## HIERARCHICAL CLUSTERING

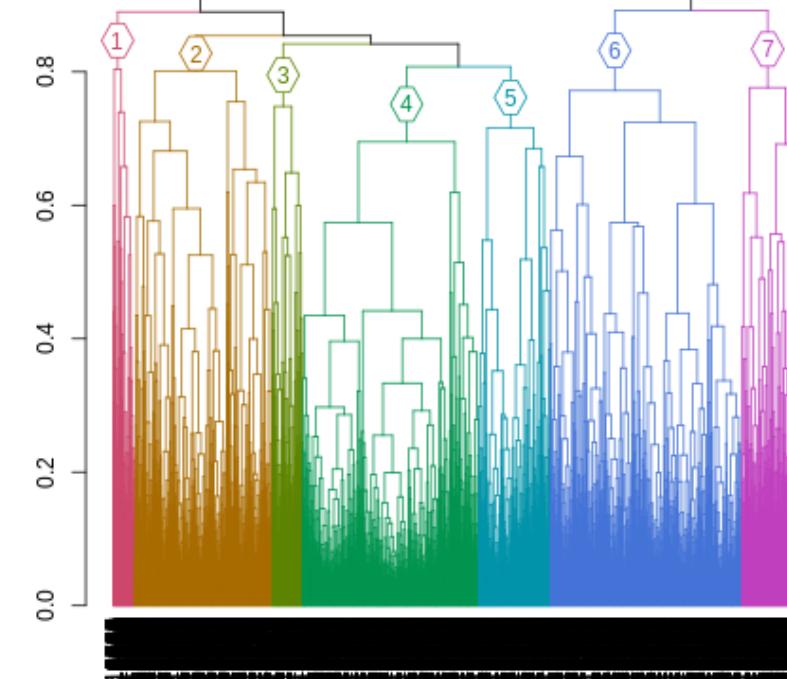
Optimal No. of Clusters



Elements per cluster

Cluster	Elements
1	2118
2	2950
3	874
4	2719
5	1121
6	474
7	335

Structural Plot



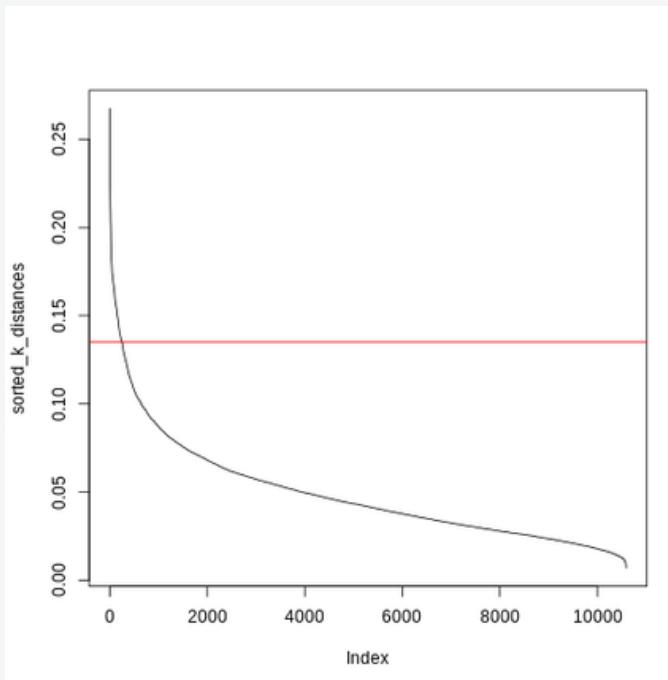
Interpretation of clusters

Cluster	GAD_T (Min-Max)	Narcissism (Min-Max)	SPIN_T (Min-Max)	Hours (Min-Max)	SWL_T (Min-Max)	Top whyplay_clean	Top Work Category	Top Playstyle_clean
1	0.0 - 21.0	1.0 - 5.0	0.0 - 64.0	0.0 - 120.0	5.0 - 35.0	fun (864)	Employed (1941)	Online w/ friends (881)
2	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	0.0 - 105.0	5.0 - 35.0	fun (936)	Students (2945)	Online w/ strangers (1537)
3	0.0 - 21.0	1.0 - 5.0	0.0 - 65.0	1.0 - 120.0	5.0 - 35.0	improving (431)	Unemployed (874)	Online w/ friends (321)
4	0.0 - 21.0	1.0 - 5.0	0.0 - 64.0	0.0 - 100.0	5.0 - 35.0	fun (1362)	Students (2296)	Online w/ friends (2719)
5	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	1.0 - 80.0	5.0 - 35.0	fun (650)	Students (1121)	Online w/ acquaintances (341)
6	0.0 - 21.0	1.0 - 5.0	0.0 - 62.0	1.0 - 100.0	5.0 - 35.0	fun (122)	Employed (75)	Singleplayer (321)
7	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	3.0 - 100.0	5.0 - 35.0	winning (275)	Students (103)	Online w/ strangers (287)

# CLUSTERING

## DBSCAN CLUSTERING

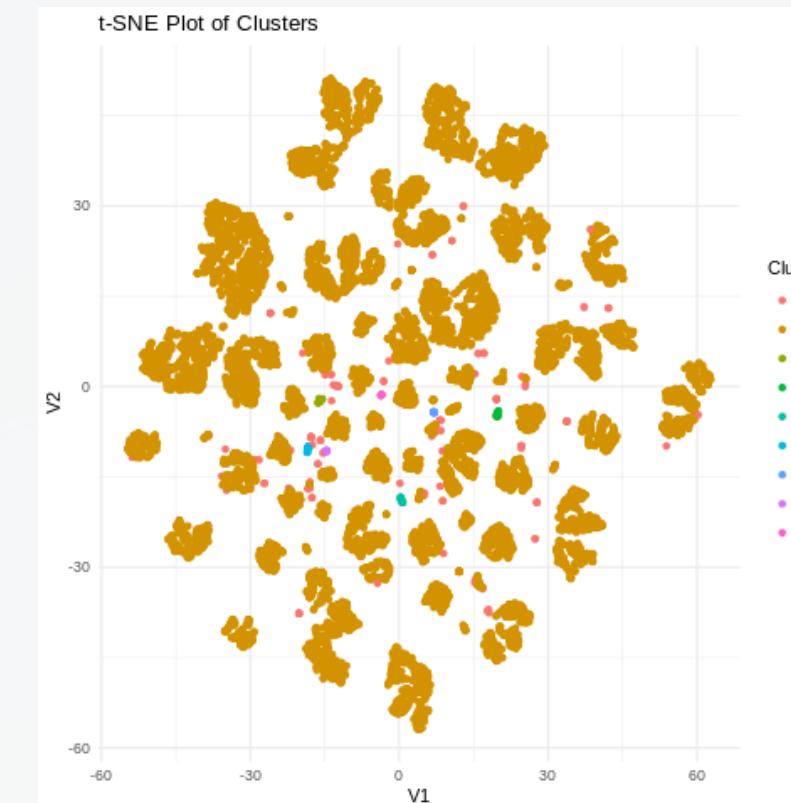
Optimal No. of Clusters



Elements per cluster

Cluster	Elements
0	116
1	10386
2	15
3	17
4	13
5	14
6	8
7	9
8	13

Structural Plot



Interpretation of clusters

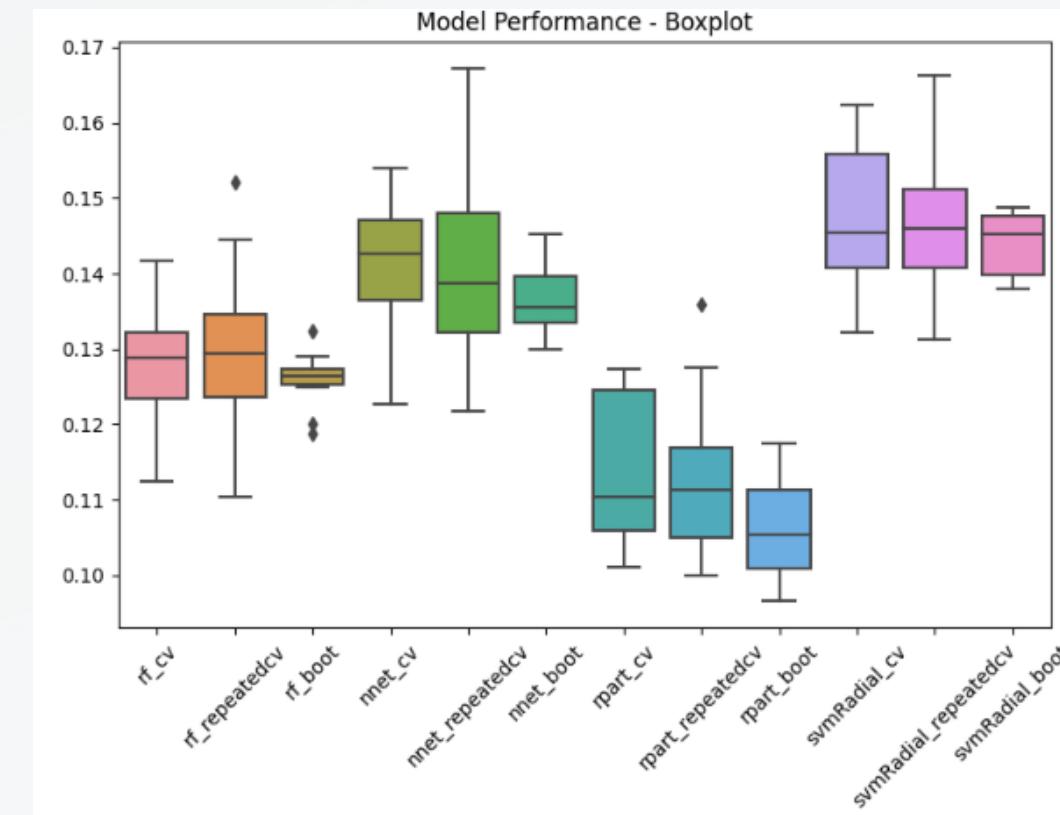
Cluster	GAD_T (Min-Max)	Narcissism (Min-Max)	SPIN_T (Min-Max)	Hours (Min-Max)	SWL_T (Min-Max)	Top whyplay_clean	Top Work Category	Top Playstyle_clean
0	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	2.0 - 120.0	5.0 - 35.0	relaxing (27)	Unemployed (43)	Online w/ acquaintances (22)
1	0.0 - 21.0	1.0 - 5.0	0.0 - 68.0	0.0 - 120.0	5.0 - 35.0	fun (4266)	Students (5654)	Online w/ RL friends (4797)
2	0.0 - 21.0	1.0 - 4.0	6.0 - 65.0	5.0 - 60.0	5.0 - 26.0	fun (15)	Unemployed (15)	Singleplayer (15)
3	0.0 - 17.0	1.0 - 4.0	7.0 - 44.0	8.0 - 35.0	10.0 - 32.0	relaxing (17)	Students (17)	Online w/ acquaintances (17)
4	0.0 - 14.0	1.0 - 4.0	2.0 - 35.0	6.0 - 87.0	9.0 - 31.0	winning (13)	Employed (13)	Singleplayer (13)
5	0.0 - 21.0	1.0 - 2.0	3.0 - 45.0	2.0 - 53.0	5.0 - 22.0	relaxing (14)	Unemployed (14)	Online w/ strangers (14)
6	0.0 - 16.0	1.0 - 3.0	7.0 - 30.0	1.0 - 20.0	10.0 - 27.0	relaxing (8)	Employed (7)	Singleplayer (8)
7	0.0 - 11.0	1.0 - 3.0	5.0 - 33.0	5.0 - 90.0	8.0 - 22.0	relaxing (9)	Unemployed (9)	Online w/ acquaintances (9)
8	2.0 - 11.0	1.0 - 3.0	0.0 - 44.0	3.0 - 35.0	10.0 - 21.0	relaxing (13)	Students (13)	Singleplayer (13)

# COMPARATIVE ANALYSIS

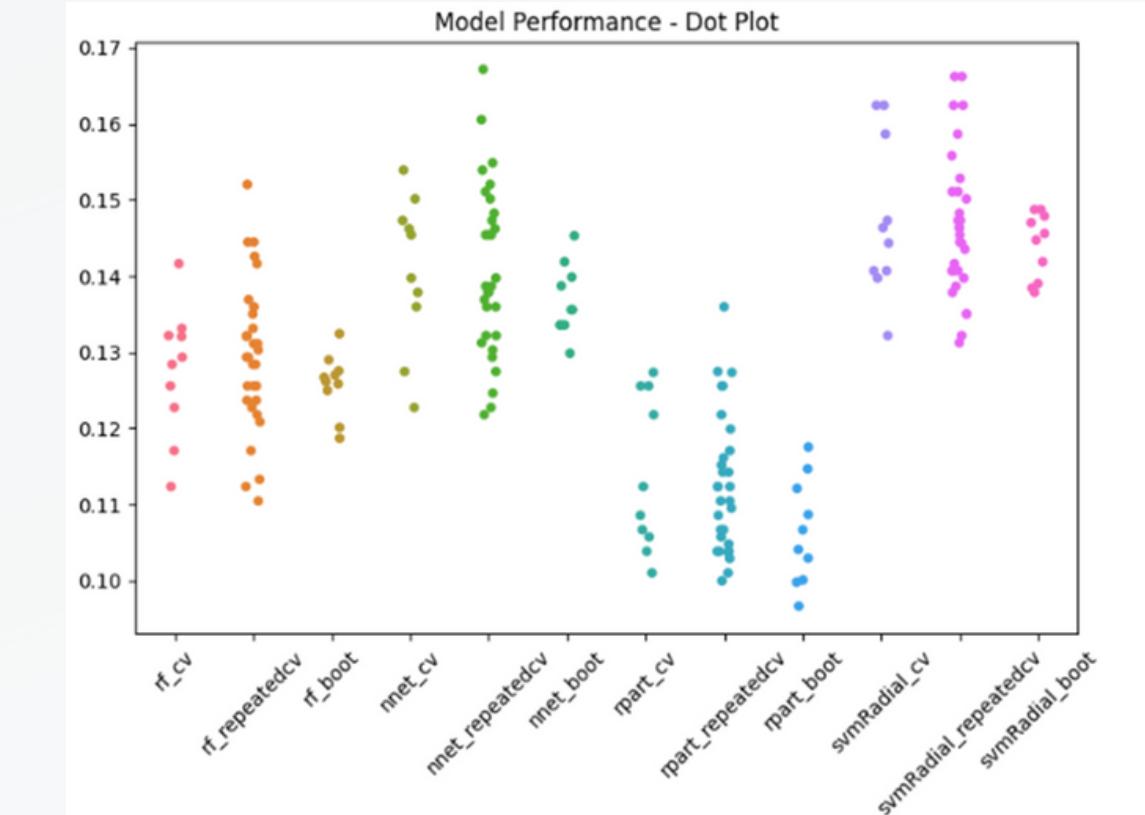
## Statistical summary

Metric	Best Model Performance	Worst Model Performance
Mean	SVM (Standard K-Fold): 14.75%	Decision Tree (Bootstrapping): 10.64%
Standard Deviation	Random Forest (Bootstrapping): 0.40%	Neural Network (Repeated K-Fold): 1.12%
Minimum	Random Forest (Bootstrapping): 11.87%	Decision Tree (Bootstrapping): 9.67%
25th Percentile	Random Forest (Bootstrapping): 12.52%	Decision Tree (Bootstrapping): 10.08%
Median (50%)	SVM (Standard K-Fold): 14.54%	Decision Tree (Bootstrapping): 10.54%
75th Percentile	SVM (Repeated K-Fold): 15.11%	Decision Tree (Standard K-Fold): 12.46%
Maximum	Neural Network (Repeated K-Fold): 16.71%	Decision Tree (Bootstrapping): 11.76%

## Box Plot



## Dot Plot



# ETHICAL ISSUES

## Overview of Dataset

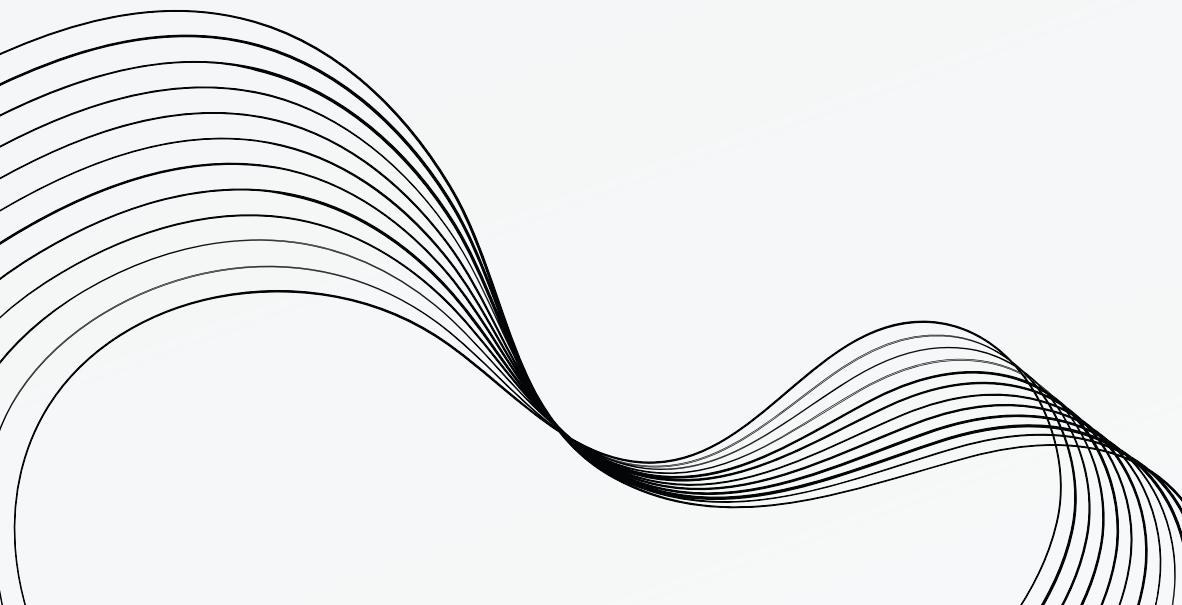
- *2017 Survey with 13,000 participants*
- *To focus on links between gaming habits, socio-economic factors, and psychological measures*

## Significance

- *Largest dataset of its kind*
- *Openly accessible*

## Procedure to address ethical issues

- *Methodology* -
  - How was the data gathered?*
  - Were the participants informed and consented?*
- *Initial Data Processing* -
  - How was the data anonymized or de-identified?*
  - Measures to protect participant privacy*



# ETHICAL ISSUES

## PROCEDURE

### Data Collection Method

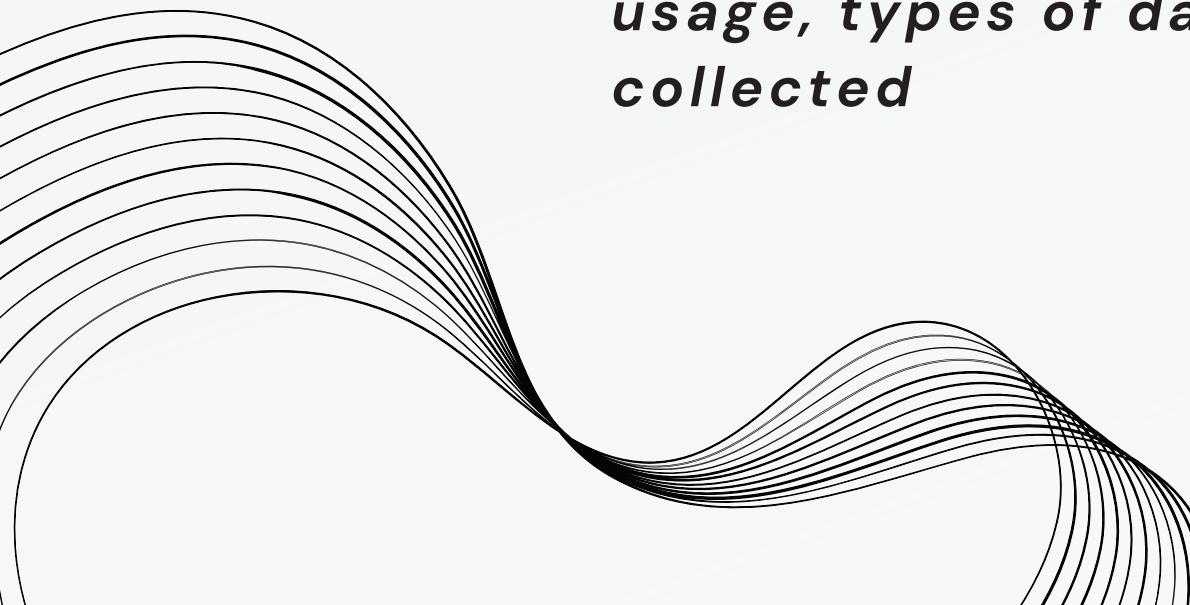
- Platforms used were Reddit, CrowdFlower, TeamLiquid.net
- Target Group was Video game players
- Information was provided about data usage, types of data collected

### Ethical Considerations

- Consent was taken from participants to ensure awareness
- Freedom was given to withdraw anytime without consequences
- No direct PII collected to maintain confidentiality

### Data Diversity

- Participants from 109 countries; majority from the USA, Germany, the UK, Canada
- Gender Imbalance exists with 12699 male, 713 female, 52 other



# ETHICAL ISSUES

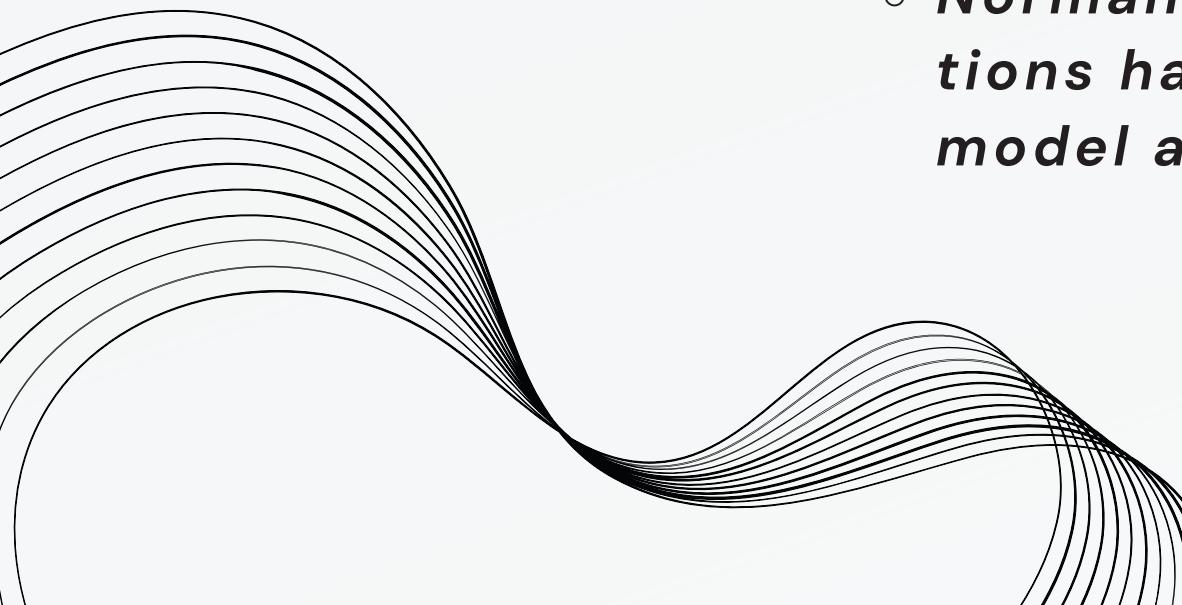
## PROCEDURE

### Data Processing and Cleaning

- *Survey responses in CSV format*
- *Outliers were removed, irrelevant columns were removed, useful categorical columns were cleaned*
- *Normalization/standardizations happened during model applications*

### Data Transformation

- *Aggregation and Segmentation happened on country, age etc. for meaningful analysis*
- *Systematic categorization of open-ended responses for research objective*



# NEURAL NETWORKS

## DATA PREPARATION

### BEFORE NORMALIZATION

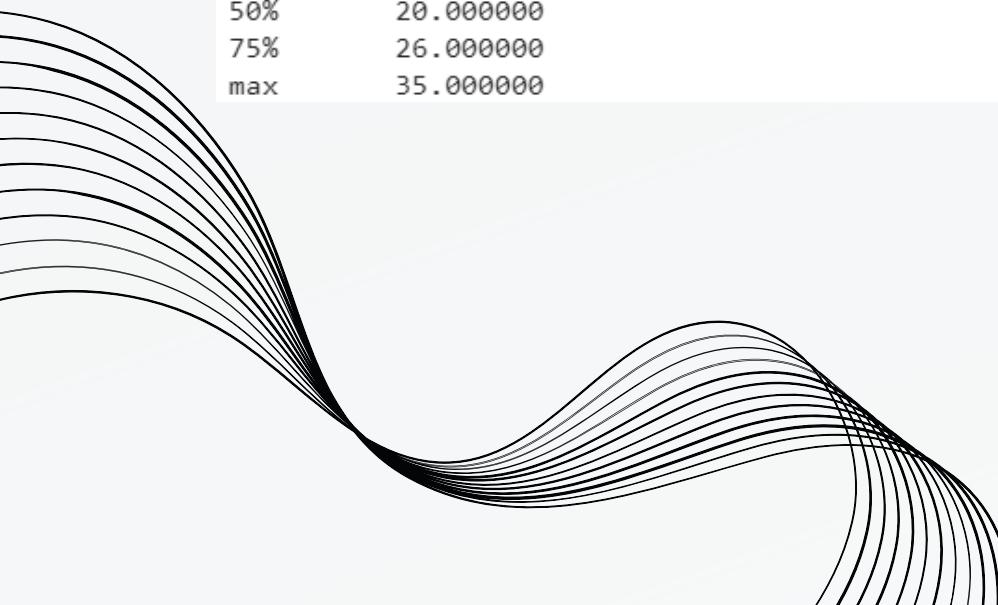
```
Before Normalization:          Age      GAD_T      Hours    Narcissism     SPIN_T  \
count  10591.000000  10591.000000  10591.000000  10591.000000  10591.000000
mean   20.828911    5.17411    21.360023    2.021433    19.730998
std    3.154145    4.66851    13.257550    1.057378    13.368407
min   18.000000    0.00000    0.000000    1.000000    0.000000
25%  18.000000    2.00000    12.000000    1.000000    9.000000
50%  20.000000    4.00000    20.000000    2.000000   17.000000
75%  22.000000    8.00000    28.000000    3.000000   28.000000
max  56.000000   21.00000   120.000000    5.000000   68.000000

           SWL_T
count  10591.000000
mean   19.838825
std    7.181354
min   5.000000
25%  14.000000
50%  20.000000
75%  26.000000
max  35.000000
```

### AFTER NORMALIZATION

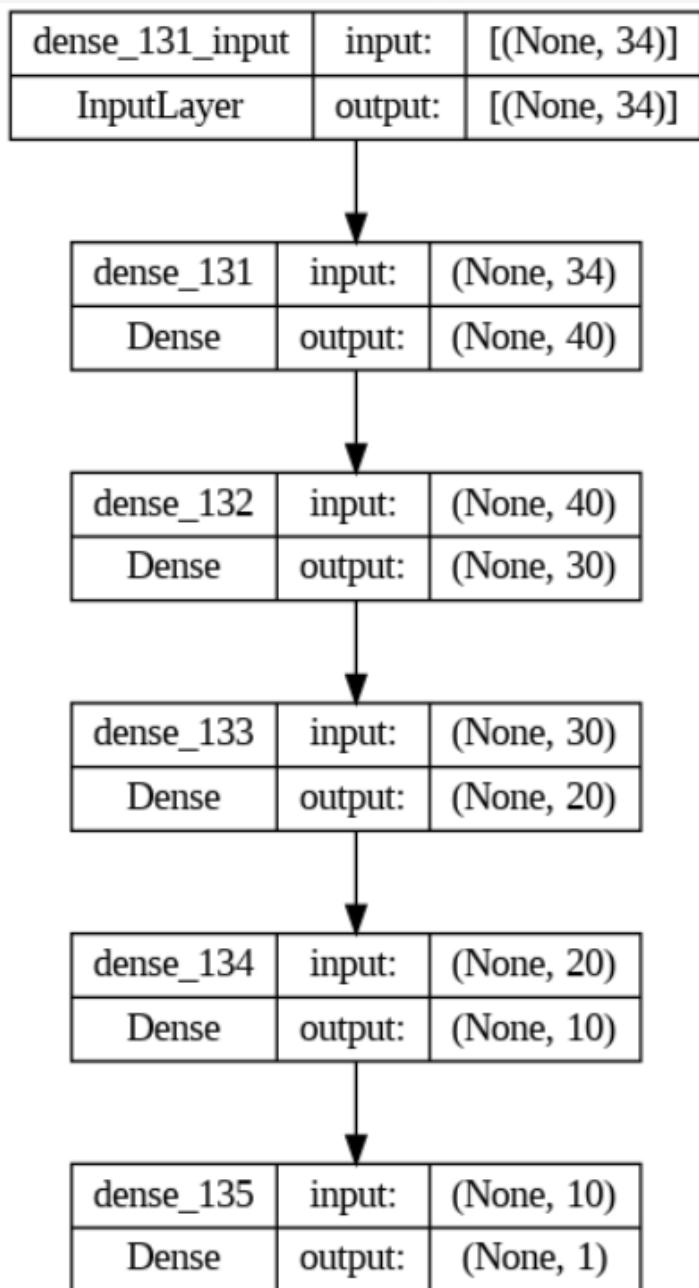
```
After Normalization:          GAD_T      Hours     SPIN_T      SWL_T  \
count  10591.000000  10591.000000  10591.000000  10591.000000
mean   0.246386    0.178000    0.290162    0.494628
std    0.222310    0.110480    0.196594    0.239378
min   0.000000    0.000000    0.000000    0.000000
25%  0.095238    0.100000    0.132353    0.300000
50%  0.190476    0.166667    0.250000    0.500000
75%  0.380952    0.233333    0.411765    0.700000
max  1.000000    1.000000    1.000000    1.000000

           whyplay_clean_all  whyplay_clean_distraction  whyplay_clean_fun  \
count  10591.000000                10591.000000            10591.000000
mean   0.007365                  0.002644            0.405344
std    0.085506                  0.051352            0.490982
min   0.000000                  0.000000            0.000000
25%  0.000000                  0.000000            0.000000
50%  0.000000                  0.000000            0.000000
75%  0.000000                  0.000000            1.000000
max  1.000000                  1.000000            1.000000
```



# NEURAL NETWORKS

## NEURAL NETWORK ARCHITECTURE DIAGRAM



## LAYER CONFIGURATIONS AND ACTIVATIONS

```
layer_configs = [(10,)- One hidden layer with 10 nodes,  
(20,)- One hidden layer with 20 nodes,  
(10, 10)- Two hidden layers with 10 nodes each,  
(20, 10)- Two hidden layers with 20 nodes in the first and 10 in the second,  
(20, 20)- Two hidden layers with 20 nodes in each,  
(30, 20, 10)- Three hidden layers with 30, 20, and 10 nodes in each  
respective layer,  
(40, 30, 20, 10)] - Four hidden layers with 40, 30, 20, and 10 in  
each respective layer.
```

```
activations = ['relu', 'sigmoid']
```

# NEURAL NETWORKS

## RESULTS AND MODEL DETAILS

```
results_reg_nn
```

```
{((10,),  
'relu'): {'loss': [0.06763207912445068,  
0.041859984397888184,  
0.03746723756194115,  
0.036340437829494476,  
0.0357864573597908,  
0.035379815846681595,  
0.035160861909389496,  
0.03506303206086159,  
0.0349060520529747,  
0.03486179932951927,  
0.034709978848695755,  
0.034604039043188095,  
0.0345996618270874,  
0.034503109753131866,  
0.03446292504668236,  
0.03434351459145546,  
0.03439820557832718,  
0.03435637801885605,  
0.03432324528694153,  
0.03415769338607788,  
0.034300852566957474,  
0.034225933253765106,  
0.03406434878706932,  
0.034058745950460434,  
0.034002020955085754,  
0.03400377184152603,  
0.03401808813214302,  
0.03397216647863388,  
0.03400079160928726,  
0.03396153822541237,  
0.03390432149171829},
```

- **EPOCHS: 100**
- **Learning Rate: 0.01**
- **With every single run of the entire dataset, for each epoch, the loss is reducing as we can see from the screenshot.**
- **We stored all the results in a dictionary.**
- **Sorted the results and printed out the best configuration with the highest mae.**

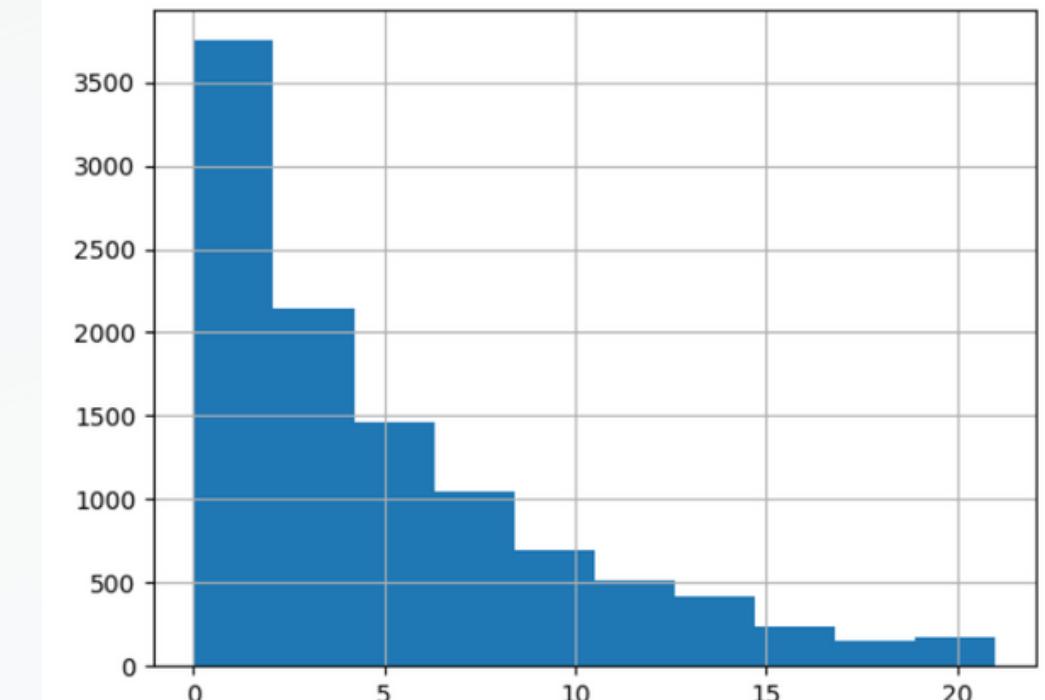
Top 5 Configurations:

```
Rank 1: Configuration ((30, 20, 10), 'sigmoid'), Best Validation Accuracy: 0.1793  
Rank 2: Configuration ((10,), 'relu'), Best Validation Accuracy: 0.1781  
Rank 3: Configuration ((10,), 'sigmoid'), Best Validation Accuracy: 0.1759  
Rank 4: Configuration ((40, 30, 20, 10), 'sigmoid'), Best Validation Accuracy: 0.1744  
Rank 5: Configuration ((10, 10), 'sigmoid'), Best Validation Accuracy: 0.1743
```

**THE HISTOGRAM SHOWS HOW  
OUR DATA IS HIGHLY SKEWED  
FOR DIFFERENT CLASSES:**

```
anxiety['GAD_T'].hist()
```

<Axes: >



# NEURAL NETWORKS

## CLASSIFICATION NEURAL NETWORK MODEL

- CATEGORIZED GAD\_T TO 2 CATEGORIES, 0(No Anxiety) AND 1(HAS ANXIETY)
- RAN THE CLASSIFICATION MODEL:

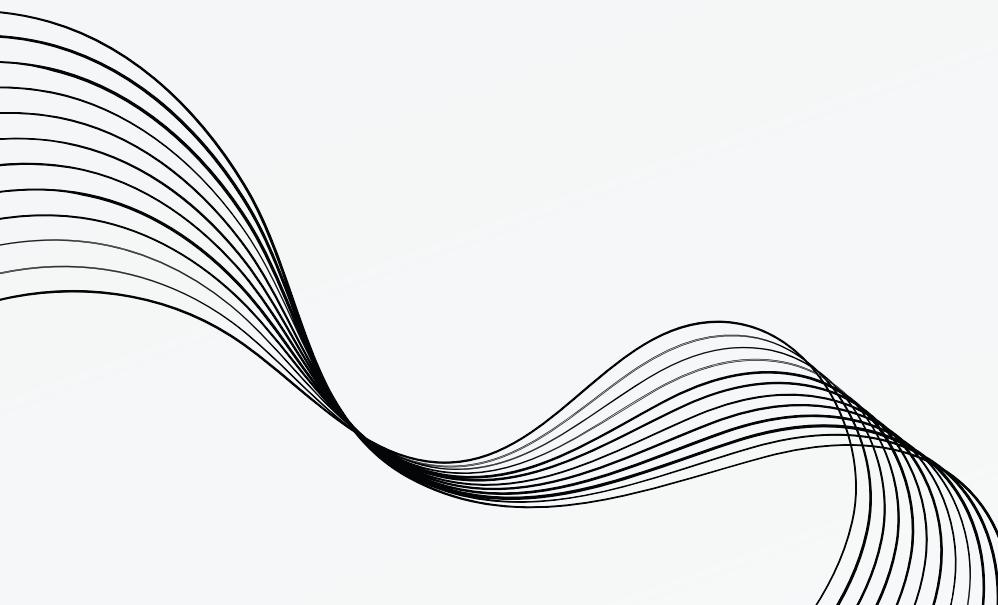
```
Best Configuration: ((40, 30, 20, 10), 'logistic')
```

```
Best Accuracy: 0.8716375648890986
```

```
Best Correlation: 0.5325210659339328
```

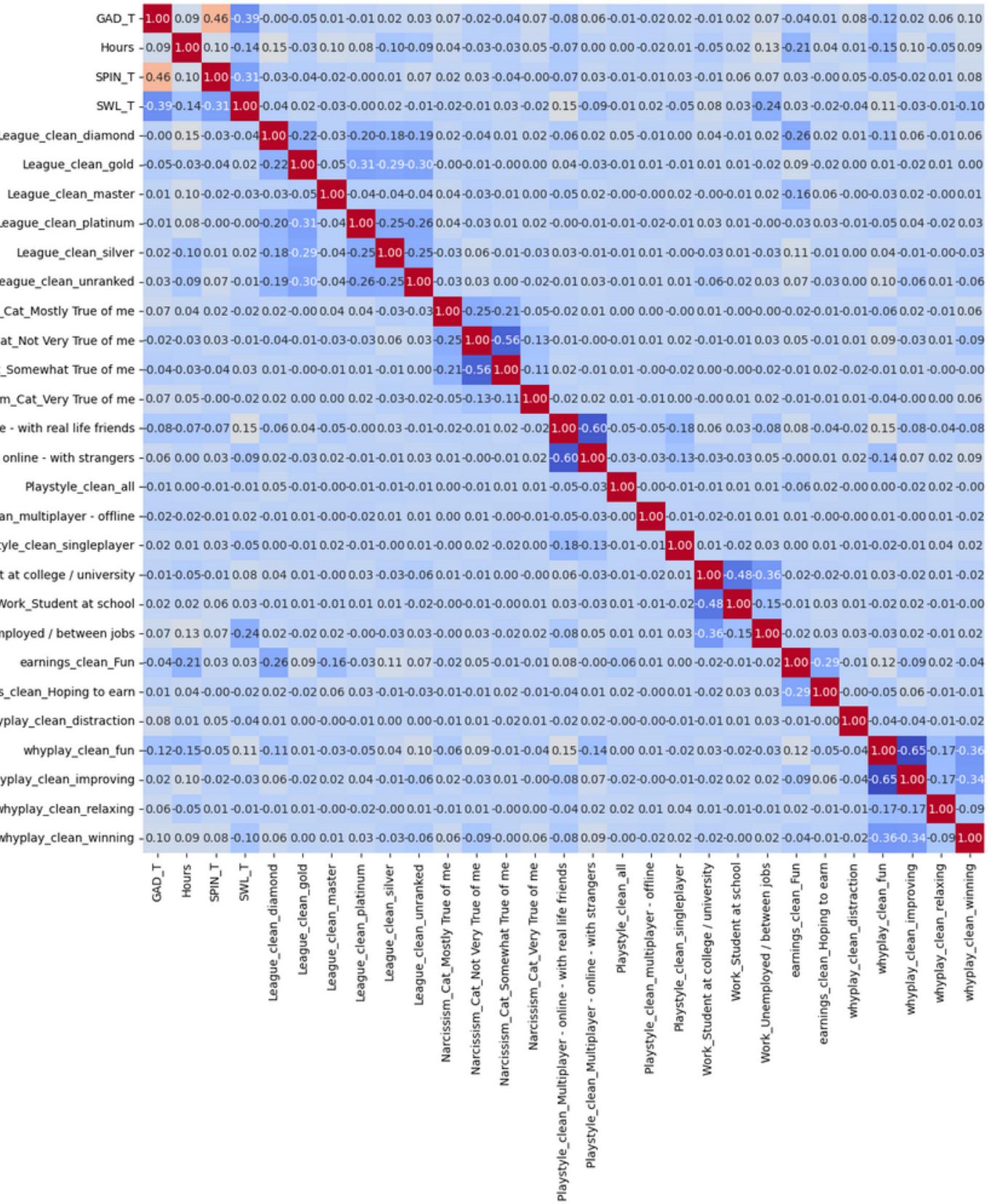
```
Confusion Matrix:
```

```
[[1767 38]
 [ 234 80]]
```



# FEATURE SELECTION

## FILTERS WITH SVM



# FEATURE SELECTION

## EMBEDDED: REGULARIZATION(L1) FOR NEURAL NETWORKS

- *L1 REGULARIZATION WITH NEURAL NETWORK FOR FEATURE SELECTION:*
- *EPOCHS: 100*
- *Learning Rate: 0.01*
- *We stored all the results in a dictionary.*
- *Sorted the results and printed out the best configuration with the highest mae.*

### OUR NEURAL NETWORK RESULTS:

Top 5 Configurations:

Rank 1: Configuration ((30, 20, 10), 'sigmoid'), Best Validation Accuracy: 0.1793  
Rank 2: Configuration ((10,), 'relu'), Best Validation Accuracy: 0.1781  
Rank 3: Configuration ((10,), 'sigmoid'), Best Validation Accuracy: 0.1759  
Rank 4: Configuration ((40, 30, 20, 10), 'sigmoid'), Best Validation Accuracy: 0.1744  
Rank 5: Configuration ((10, 10), 'sigmoid'), Best Validation Accuracy: 0.1743

### WITH REGULARIZATION:

Top 5 Configurations:

Rank 1: Configuration ((10,), 'sigmoid'), Best Validation Accuracy: 0.1928  
Rank 2: Configuration ((20, 20), 'sigmoid'), Best Validation Accuracy: 0.1874  
Rank 3: Configuration ((20,), 'sigmoid'), Best Validation Accuracy: 0.1849  
Rank 4: Configuration ((20, 20), 'relu'), Best Validation Accuracy: 0.1847  
Rank 5: Configuration ((30, 20, 10), 'sigmoid'), Best Validation Accuracy: 0.1842

# FEATURE SELECTION

## WRAPPER: SFS WITH LINEAR REGRESSION

### SHORTLISTED VARS:

```
> print(shortlistedVars)
[1] "(Intercept)"
[3] "SWL_T"
[5] "whyplay_cleanfun"
[7] "whyplay_cleanrelaxing"
[9] "Narcissism_CatMostly True of me"
[11] "Narcissism_CatSomewhat True of me"
[13] "League_cleandiamond"
[15] "League_cleanmaster"
[17] "League_cleansilver"
[19] "earnings_cleanFun"
[21] "workStudent at college / university"
[23] "workUnemployed / between jobs"

[1] "SPIN_T"
[3] "whyplay_cleandistraction"
[5] "whyplay_cleanimproving"
[7] "whyplay_cleanwinning"
[9] "Narcissism_CatNot Very True of me"
[11] "Narcissism_CatVery True of me"
[13] "League_cleangold"
[15] "League_cleanplatinum"
[17] "League_cleanunranked"
[19] "earnings_cleanHoping to earn"
[21] "workStudent at school"
[23] "Hours"
```

### MODEL SUMMARY:

```
lm(formula = GAD_T ~ SPIN_T + SWL_T + whyplay_clean + Narcissism_Cat +
  League_clean + earnings_clean + Work + Hours, data = dataforSFS)

Residuals:
    Min      1Q   Median      3Q     Max 
-11.3310 -2.6256 -0.6518  1.8955 20.5710 

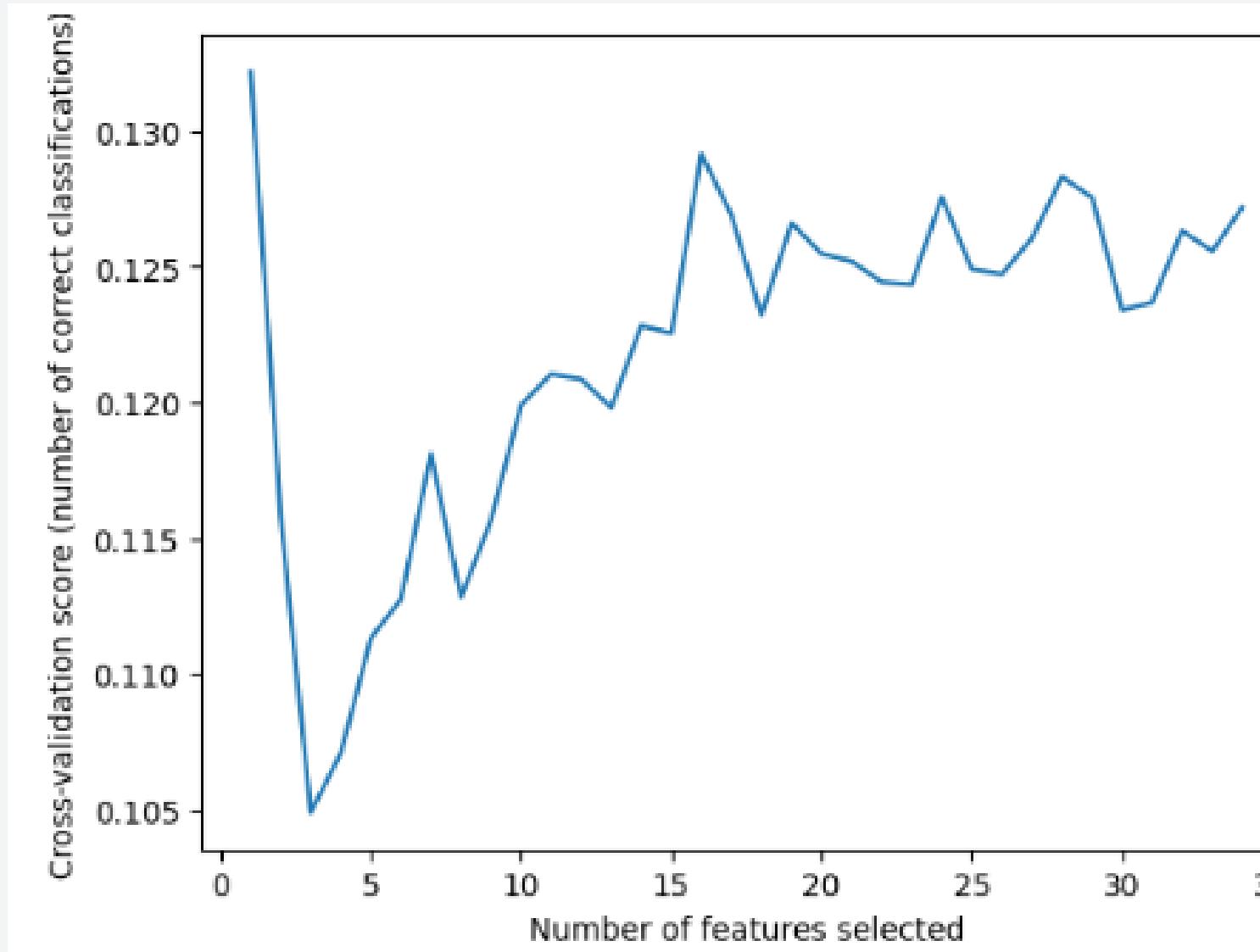
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 9.863388  1.402301  7.034 2.14e-12 ***
SPIN_T       0.125444  0.003043 41.224 < 2e-16 ***
SWL_T        -0.176996 0.005790 -30.572 < 2e-16 ***
whyplay_cleandistraction 3.731554  0.865384  4.312 1.63e-05 ***
whyplay_cleanfun      -0.710103  0.448490 -1.583 0.11338  
whyplay_cleanimproving -0.304068  0.448629 -0.678 0.49793  
whyplay_cleanrelaxing   0.642749  0.481473  1.335 0.18192  
whyplay_cleanwinning   0.043807  0.454999  0.096 0.92330  
Narcissism_CatMostly True of me 0.666192  0.159269  4.183 2.90e-05 ***
Narcissism_CatNot Very True of me -0.264509 0.111332 -2.376 0.01753 *  
Narcissism_CatSomewhat True of me -0.192584 0.114002 -1.689 0.09119 .  
Narcissism_CatVery True of me    1.746655  0.266548  6.553 5.91e-11 ***
League_cleandiamond   -3.245781 1.313055 -2.472 0.01345 *  
League_cleangold      -2.995952 1.309847 -2.287 0.02220 *  
League_cleanmaster    -2.845912 1.379940 -2.062 0.03920 *  
League_cleanplatinum -2.955779 1.310444 -2.256 0.02412 *  
League_cleansilver    -2.490981 1.310401 -1.901 0.05734 .  
League_cleanunranked -2.680360 1.310277 -2.046 0.04082 *  
earnings_cleanFun     -0.703626 0.150867 -4.664 3.14e-06 ***
earnings_cleanHoping to earn -0.494799 0.443069 -1.117 0.26413  
workStudent at college / university 0.275380 0.100839  2.731 0.00633 ** 
workStudent at school      0.173412 0.127854  1.356 0.17502  
workUnemployed / between jobs -0.035553 0.153667 -0.231 0.81704  
Hours                  0.004823 0.003056  1.578 0.11454  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.92 on 10567 degrees of freedom
Multiple R-squared:  0.2965,    Adjusted R-squared:  0.2949 
F-statistic: 193.6 on 23 and 10567 DF,  p-value: < 2.2e-16
```

# FEATURE SELECTION

## HYBRID: RFE WITH RANDOM FORESTS

```
RFEcv
+ estimator: RandomForestClassifier
  + RandomForestClassifier
```



Optimal number of features : 1  
Index(['SPIN\_T'], dtype='object')



**THANK YOU**