SOCR 2021 MDP Project Summaries

The one-page summaries below describe the main SOCR MDP R&D Projects for 2021 (January-December)

GDrive: [https://drive.google.com/drive/folders/1OZFGWy1xtHcAaZPeOaYMjW6oe9HYz8E1](https://drive.google.com/drive/folders/1OZFGWy1xtHcAaZPeOaYMjW6oe9HYz8E1)

GSlides: [https://docs.google.com/presentation/d/1Ntw3d-yPcGrRLi1puK8uKCy8bWDi3D5Buuff8jHVYEo/](https://docs.google.com/presentation/d/1Ntw3d-yPcGrRLi1puK8uKCy8bWDi3D5Buuff8jHVYEo/)

SOCR Project Leaders:
- Programming: Alex Kalinin, Ivo Dinov
- Methods (CBDA & DataSifter): Simeone Marino & Nina Zhou
- Analytics: Jared Chai, Nina Zhou, Ivo Dinov
- Spacekime Analytics: Daxuan, Rongqian, Yueyang, Ivo Dinov

SOCR Trainees/Students: [https://docs.google.com/spreadsheets/d/1cgn4bBSKHY_nTPPHznWgHxtWsbh0rCy0pE6Kdm7y9nc](https://docs.google.com/spreadsheets/d/1cgn4bBSKHY_nTPPHznWgHxtWsbh0rCy0pE6Kdm7y9nc)

<table>
<thead>
<tr>
<th>Project Summaries</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Area</strong></td>
</tr>
<tr>
<td>Programming Subteam: <strong>SOCRAT</strong> (Charts, Wrangler, Modeler, Analyses, Tools) (2-3 students)</td>
</tr>
<tr>
<td>Methods (CBDA &amp; DataSifter) (4 students) <strong>DataSifter &amp; CBDA</strong></td>
</tr>
<tr>
<td>Analytics (4 students) <strong>TDA Biomed/Health Applications</strong> (see Case-Studies)</td>
</tr>
<tr>
<td>Spacekime Analytics (sub-team – working directly with the PI) (4 students) <a href="https://www.spacekime.org">www.spacekime.org</a></td>
</tr>
</tbody>
</table>

**SOCR Computing servers:**
- ARC-TS: [https://arc-ts.umich.edu/open-ondemand/](https://arc-ts.umich.edu/open-ondemand/)
- SOCR-pipeline: socr-pipeline.nursing.umich.edu
- SOCR-RShiny: rshiny.umms.med.umich.edu
- SOCR-Lighthouse: [https://lighthouse.arc-ts.umich.edu](https://lighthouse.arc-ts.umich.edu) (Lighthouse User Guide)

[https://www.socm.umd.edu/](https://www.socm.umd.edu/)
SOCR 2021 MDP Project: SOCRAT

SOCR Project Leaders: Alex Kalinin / Ivo Dinov
Website: http://socr.umich.edu/HTML5/SOCRAT/
GitHub: https://github.com/SOCR/SOCRAT
Training Modules: https://github.com/SOCR/socr-tutorials
GDrive: https://drive.google.com/drive/folders/1UrNpNDI5sWoXW61YwP02NSv3PBbxfyvP

Description
The Statistics Online Computational Resource Analytics Toolbox (SOCRAT) is a Dynamic Web Toolbox for Interactive Data Processing, Analysis, and Visualization. It’s purely built using HTML5 standards and JavaScript (core library) as well as node.js,

Student Skills
● EECS, Computer Science (CSE/CS-LSA) and School of Information (SI)
● UI/UX design, HTML5, JavaScript

Project Goals
● Go through the Training Modules, practice HTML/JS/Angular/Node programming
● Get your GitHub domain going and pull current SOCRAT branch
● Choose 1-2 deliverables, go over current design, start expansion, include unit tests, pilot development
● Coordinate with team

Deliverables
● Expanded collection of Charts
● Expanded collection of Data-Modelers
● Expanded collection of (parametric and non-parametric) Statistical Analyses
● Expanded collection of machine learning classification, prediction, clustering and analytics modules.

Team Activities
● Weekly team BlueJeans meetings
● Code review (pull/push Github requests)
● Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

References
● Review the websites

https://www.socr.umich.edu/html/SOCR_Research.html
SOCR 2021 MDP Project: **Methods: CBDA**

**SOCR Project Leaders:** Simeone Marino

**Website:** [http://socr.umich.edu/HTML5/CBDA/](http://socr.umich.edu/HTML5/CBDA/)

**GitHub:** [https://github.com/SOCR/CBDA](https://github.com/SOCR/CBDA)

**C-RAN Package:** [https://cran.r-project.org/web/packages/CBDA](https://cran.r-project.org/web/packages/CBDA)

**Training Modules:** [http://socr.umich.edu/HTML5/CBDA/](http://socr.umich.edu/HTML5/CBDA/)

**GDrive:** [https://drive.google.com/drive/folders/1hjwtgz64A_IUsnRK1qv7mGSJ3HdBHaRW](https://drive.google.com/drive/folders/1hjwtgz64A_IUsnRK1qv7mGSJ3HdBHaRW)

**Description**

The SOCR Compressive Big Data Analytics (CBDA) Project conducts research and implements efficient computational algorithms to tackle the Big Data problems of representation and analysis of complex heterogeneous information. Big Data cannot be loaded and processed as a whole. CBDA implements a real-time efficient divide-and-conquer strategy to deconstruct the Big Data into meaningful pieces of information that can be eventually reconstructed for actionable knowledge and predictive analytics.

**Student Skills**

- Probability, stats, math, numerical methods, optimization
- R programming with RStudio (IDE) experience

**Project Goals**

- Go through the provided materials and references
- Download the CBDA Package
- Practice with test-cases ([https://umich.instructure.com/courses/38100/files/folder/Case_Studies](https://umich.instructure.com/courses/38100/files/folder/Case_Studies))
- Identify specific R&D direction to go deeper into an meaningfully contribute to CBDA
- Coordinate with team

**Deliverables**

- New CBDA methods
- Expanded collection of machine learning forecasting, prediction, classification, clustering methods to expand the available CBDA algorithms
- Release new versions of CBDA R package and publish CBDA #2 manuscript
- Python/Perl scripts to speed up the subsampling strategy with Big Data > 100Gb-1Tb

**Team Activities**

- Weekly team face-to-face/BlueJeans meetings
- Code review (pull/push Github requests)
- Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

**References**

- Review the websites

[https://www.socr.umich.edu/html/SOCR_Research.html](https://www.socr.umich.edu/html/SOCR_Research.html)
The SOCR DataSifter is a novel method, and an efficient R package, for on-the-fly de-identification of structured Clinical/Epic/PHI data. This approach provides complete administrative control over the risk of data identification when sharing large clinical cohort-based medical data. At the extremes, the data-governor may specify that either null data or completely identifiable data is generated and shared with the data-requester. This decision may be based on data-governor determined criteria about access level, research needs, etc. For instance, to stimulate innovative pilot studies, the data office may dial up the level of protection (which may naturally devalue the information content in the data), whereas for more established and trusted investigators, the data governors may provide a more egalitarian dataset that balances preservation of information content and sensitive-information protection.

Student Skills
- Probability, stats, math, numerical methods, optimization
- R programming with RStudio (IDE) experience

Project Goals
- Go through the provided materials and references
- Download the DataSifter-lite Package
- Practice with test-cases (https://umich.instructure.com/courses/38100/files/folder/Case_Studies)
- Identify specific R&D direction to go deeper into and meaningfully contribute to DataSifter methods, implementation and/or validation
- Coordinate with team

Deliverables
- New DataSifter methods/algorithms (e.g., addressing text, time-varying, graph data organizations)
- Release new versions of DataSifter R package
- Coordinate/support collaborators

Team Activities
- Weekly team face-to-face/BlueJeans meetings
- Code review (pull/push Github requests)
- Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

References
- Review the websites
SOCR 2021 MDP Project: Webapps & Data Analytics

SOCR Project Leaders: Jared Chai, Simeone Marino, Ivo Dinov

Website: <many, e.g., http://socr.umich.edu/HTML5>
GitHub: https://github.com/SOCR <many, e.g., https://github.com/SOCR/ALS_PA>
Training Modules: http://DSPA.predictive.space
GDrive: https://drive.google.com/drive/folders/1sN1fLYA0oLf1L4e1REJRthaMD0jXBs7w

Description
The SOCR Webapp & Data Analytics projects are focused on interrogating massive amounts of complex biomedical and health data. Each project tackles multiple case-studies using R/RMD/RStudio, RShiny Services, and Python/Jupyter Notebook and the SOCR-Flux Compute Server (https://docs.google.com/document/d/1UmBq_BMiMeUcijvKUCzPeG3tKZaWkinVtKrVWenPK1Y). The webapp development will use R markdown notebook, RShiny web-servers, and Google BigQuery Datasets.

Student Skills
● Biostats, quantitative analytics, probability, stats, math, numerical methods, optimization
● R programming with RStudio (IDE) experience, and/or Python/Jupyter Notebook

Project Goals
● Go through the provided materials and references
● Review the SOCR Data Analytics Publications (http://socr.umich.edu/people/dinov/publications.html)
● Review the SOCR R-environment (https://drive.google.com/file/d/1-u9adsMIYmMkPD9W_6BbfC1IMETsHF/)
● Practice with test-cases (https://umich.instructure.com/courses/38100/files/folder/Case_Studies)
● Identify specific case-study and an R&D direction to go deeper into an meaningfully contribute
● Coordinate with team

Deliverables
● New SOCR end-to-end data analytics protocols
● Analytical results, abstracts, publications, presentations, research findings, etc.
● MIMIC-III analytics
● Baby-growth and mother-obesity relations
● Data Value Metric (DVM)
● European Economics Indicators (longitudinal analytics)
● 2D, 3D, 4D Visualization of complex data
● Coordinate/support collaborators

Team Activities
● Weekly team face-to-face/BlueJeans meetings
● Code review (pull/push Github requests)
● Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

References
● Review the websites and listed resources
  https://shiny.med.umich.edu/apps/dinov/RShinyApp_PIPM/
  https://socr.shinyapps.io/RShinyApp_PIPM/

https://www.socr.umich.edu/html/SOCR_Research.html
SOCR 2021 MDP Project: Data Analytics - MIMIC-III

SOCR Project Leaders: Jared Chai, Simeone Marino, Ivo Dinov

Website: TBD
GitHub: https://github.com/SOCR

Training Modules:
- Data Science & Predictive Analytics: http://DSPA.predictive.space
- Previous SOCR Data Analytics Publications: http://socr.umich.edu/people/dinov/publications.html
- Gaining access to the dataset requires an online training module; see onboarding materials below https://drive.google.com/drive/u/1/folders/1Y6Yqq1CuTkHq5rZg-C9r8_je18nM886l

GDrive: https://drive.google.com/drive/folders/1sN1fLYA0oLf1I4e1REJRthaMD0jXBs7w

Description
This SOCR Data Analytics project is focused on interrogating the MIMIC-III database, a large collection of ~43,000 critical care patients from an ICU in Boston, MA. We will use R/RStudio, Python/Jupyter, and the SOCR-Flux Compute Server\(^1\) to digest the vital signs, laboratory results, free-text data, and waveforms available in this unique dataset and predict clinical outcomes via statistical modeling tools.

\(^1\)SOCR-Flux Compute server: https://docs.google.com/document/d/1UmBq_BMiMeUcijvKUCzPeG3tKZaWkinVtKrVWenPK1Y

Student Skills
- Biostats, quantitative analytics, probability, stats, math, numerical methods
- Programming experience in R (with RStudio) or Python (with Jupyter Notebook)
- Relational databases & structured query language (SQL)

Project Goals
- Review the provided materials and references (see above)
- Request access to the MIMIC-III dataset (https://mimic.physionet.org/gettingstarted/access/)
  - This involves an online but comprehensive human subjects research ethics course
- Practice with demo dataset (https://physionet.org/works/MIMICIIIClinicalDatabaseDemo/) and the MIMIC Query Builder (https://querybuilder-lcp.mit.edu/dashboard.cgi)
- Identify specific research aims and questions of interest to the team
- Coordinate with team to create a reproducible, accessible answer to these specific aims

Deliverables
- New SOCR end-to-end data analytics protocols
- Data extraction & time-alignment tools for the MIMIC-III dataset
- Build statistical models to predict meaningful clinical outcomes
- Analytical results, abstracts, publications, presentations, research findings, etc.
- Visualization of complex, multidimensional data

Team Activities
- Weekly team face-to-face/BlueJeans meetings
- Code review (pull/push Github requests)
- Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)
**SOCR 2021 MDP Project: SOCR TensorFlow/TensorBoard Apps**

**SOCR Project Leader:** Alex Kalinin, Chiranjeevi Vegi <vegi@umich.edu>, Ivo Dinov

**Website:** [http://socr.umich.edu/HTML5/SOCR_TensorBoard_UKBB](http://socr.umich.edu/HTML5/SOCR_TensorBoard_UKBB)

**GitHub:** [https://github.com/SOCR/97-tensorflowjs-quick-start](https://github.com/SOCR/97-tensorflowjs-quick-start)

**Training Modules:** [https://js.tensorflow.org/tutorials/](https://js.tensorflow.org/tutorials/)

**GDrive:** [https://drive.google.com/drive/folders/1wJY8539tpLmYiJc_vKZvI6oDVDAHTQu9](https://drive.google.com/drive/folders/1wJY8539tpLmYiJc_vKZvI6oDVDAHTQu9)

**Description**

The SOCR TensorFlowJS/TensorBoardJS project aims to design, build, validate and release new webapps based on the ML TensorFlow framework. For example, students will dive deep into TensorFlow.JS ([https://js.tensorflow.org](https://js.tensorflow.org), [https://js.tensorflow.org/api/latest/](https://js.tensorflow.org/api/latest/), [https://codepen.io/pen/?&editors=1011](https://codepen.io/pen/?&editors=1011)) and TensorBoard.JS ([https://github.com/tensorflow/tensorboard](https://github.com/tensorflow/tensorboard), [https://www.tensorflow.org/guide/summaries_and_tensorboard](https://www.tensorflow.org/guide/summaries_and_tensorboard)).

**Student Skills**

- EECS, Computer Science (CSE/CS-LSA) and School of Information (SI)
- AngularJS, TensorFlowJS, TensorBoard, JavaScript, HTML5

**Project Goals**

- Go through the Training Modules, practice HTML/JS/Angular/Node programming
- Get your GitHub domain going and start pilot testing various applications
- Use SOCR Data to experiment
- Review Vegi’s SOCR t-SNE TensorFlow Webapp ([http://socr.umich.edu/HTML5/SOCR_TensorBoard_UKBB](http://socr.umich.edu/HTML5/SOCR_TensorBoard_UKBB))
- Coordinate with team
- Rapid RDD (research, development and deployment) is needed in this project

**Deliverables**

- 2-5 new SOCR TF/TB Apps
- ...

**Team Activities**

- Weekly team BlueJeans meetings
- Code review (pull/push Github requests)
- Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

**References**

- Review the websites
SOCR 2021 MDP Project:
Interactive Graphical Probability Distribution Calculator

SOCR Project Leader: Jared Chai, Ivo Dinov

Website: http://Distributome.org &
https://shiny.med.umich.edu/apps/dinov/SOCR_DistribCalc_RShiny_App/
GitHub: https://github.com/distributome
GDrive: https://drive.google.com/drive/folders/184p8VNSOumYEG_SOxlo4MyLVtang9xLY

Goal: Stand-alone RMD source and a demo HTML (RMD-output) that address the goal of the challenge, provide the desired functionality, and implement it efficiently, without any back-end support (e.g., no shiny server apps).
Deliverables: Stand-alone RMD source and a demo HTML (RMD-output) that address the goal of the challenge, provide the desired functionality, and implement it efficiently.

Background:
- Review DSPA Chapter 2 (for probability distributions) and Chapter 5 (for plot_ly demos)
- See this DSPA Interactive Normal Probability CDF calculation tool.
- Review, experiment and play with the Probability Distributome Calculators. Try at least 2-dozen distributions - what works well and what can be improved there? Mind the selection of parameters and the choice for function to plot (PDF, CDF).

Desired Functionality: The schematic below illustrates the core functionality of the interactive probability distribution calculator. Be creative in your solution.

- Include a drop-down list for the user to select the distribution
- Include an effective strategy to specify the parameters of the selected distribution, mind that the parameter number, interpretation and values will be different for different distributions.
- Make sure you keep the interactive aspect of the interface (plot_ly style interactivity)
- Make sure all axes are appropriately scaled, labeled and drawn.

Functionality Annotations
- A: x and y axes ranges and labels
- B: Appropriate title (placed to avoid overlaps
- C: Selection of the specific distribution - should include at least 20 distributions, see class notes the Distributome Calculators.
- D: appropriate section of the specific distribution parameters

https://www.socr.umich.edu/html/SOCR_Research.html
E: cut off of the critical value (Z)
F: Ranges of the animation slider should match the x-axis range (Z-values range)
G: Play button and the animation point provide the manual user control over the critical value cut off
H: Report the appropriate Z-values, density curve height, and cumulative distribution up to Z (i.e., P(X<Z))
J: There should be a light-colored vertical line at the animation index == Z-value and extending up to the corresponding density height
K: shaded area represents the integral CDF value ...
L: Drop-down selected for plotting PDF, CDF or inverse-CDF (quantile) function to plot.

Starting R Code: The basic skeleton of one solution (using "plot_ly") is included below. Many solutions are possible and you can start with anything you like, including this initial script.

```r
library(magrittr); library(plotly)
select the right user-specified distribution (drop down list)
# Assuming Std Normal N(0,1) going down
# define the range
z<-seq(-4, 4, 0.1)
# points from -4 to 4 in 0.1 steps
# Define the quantile levels for the inverse-CDF (quantile) function
q<-seq(0.001, 0.999, 0.01)
# probability quantile values from 0.1% to 99.9% in 0.1% steps
# define a DF containing Z, PDF and CDF
dStandardNormal <- data.frame(Z=z, Density=dnorm(z, mean=0, sd=1), Distribution=pnorm(z, mean=0, sd=1))
# define an index feature
dStandardNormal$ID <- seq.int(nrow(dStandardNormal))
# Aggregate frames for interactive plot
aggregate_by <- function(dataset, feature) {
  feature <- lazyeval::f_eval(feature, dataset)
  levels <- plotly:::getLevels(feature)
  aggData <- lapply(seq_along(levels), function(x) {
    cbind(dataset[feature %in% levels[seq(1, x)]], frame = levels[[x]])
  })
  dplyr::bind_rows(aggData)
}
# Apply the aggregate to ID index
dStandardNormal <- dStandardNormal %>% aggregate_by(~ID)
# generate the Plot_ly object
plotMe <- dStandardNormal %>% plot_ly( x = ~Z, y = ~Density, frame = ~frame, type = 'scatter', mode = 'lines', fill = 'tozerox', fillcolor="red", line = list(color = "blue"), text = ~paste("Z: ", Z, " Density: ", Density, "CDF: ", Distribution), hoverinfo = 'text' ) %>%
layout( title = "Standard Normal Distribution Distribution", yaxis = list( title = "N(0,1) Density", range = c(0,0.45), zeroline = F, tickprefix = "" ) %>%
  animation_opts( frame = 100, transition = 1, redraw = FALSE ) %>%
  animation_slider( currentvalue = list( prefix = "Z: " ) ) %>%
# display interactive plot
plotMe
```

The optimal solution will include RMD (source) and HTML output (webapp).

https://www.socr.umich.edu/html/SOCR_Research.html
SOCR 2021 MDP Project:
Data Science Fundamentals: Spacekime Analytics

SOCR Project Leader: Daxuan, Yueyang, Rongqian, Milen Velev, Ivo Dinov

Website: http://tciu.predictive.space, www.spacekime.org
GitHub: https://github.com/SOCR/TCIU
GDrive: https://drive.google.com/drive/folders/1PMMBR2bzBPubYMpywLkcTkJPyxOKQ4Aq

Description
The SOCR Data Science Fundamentals project will explore new theoretical representation and analytical strategies to understand large and complex data. It will utilize information measures, entropy KL divergence, PDEs, Dirac’s bra-ket operators. This fundamentals of data science research project will explore time-complexity and inferential uncertainty in modeling, analysis and interpretation of large, heterogeneous, multi-source, multi-scale, incomplete, incongruent, and longitudinal data.


Student Skills
● Physics, math or engineering background
● R programming with RStudio (IDE) experience, and/or Python/Jupyter Notebook

Project Goals
● Go through the provided materials and references
● Review the current platform (will be provided)
● Perform 3D and 4D Plot_Ly visualization of complex manifolds, including 5D space-kime and 2D-curved Kime.
● Identify specific case-study and an R&D direction to go deeper into an meaningfully contribute
● Coordinate with team

Deliverables
● Visualization protocols
● Math proofs of various physics properties in 5D Minkowski spacekime

Team Activities
● Weekly team face-to-face/BlueJeans meetings
● Code review Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

Key points
● What is the problem? Use complex-time physics to formulate data science theory & practice
● Why is it important? There is currently no canonical theory for Big Data discovery science
● What is the SOCR Solution? Blend transdisciplinary knowledge to build a new Data Analytic method
● It’s real; here it is (in a pilot form) … demo … See TCIU Video
● Why should you consider joining this SOCR-MDP Project? High-risk/high-potential yield project.

References
● Review the websites and listed resources
● TCIU Website: http://tciu.predictive.space/
● TCIU GitHub: https://github.com/SOCR/TCIU/
● www.SpaceKime.org
SOCR 2021 MDP Project: Synthetic Data

SOCR Project Leaders: Simeone Marino, Johnny Liu, Ronak Shetty, Ivo Dinov
Website: http://gray-rain.com
GitHub: ...
Training Modules: https://github.com/SOCR/socr-tutorials
GDrive: https://drive.google.com/drive/folders/1hvzoihZMO-6uJlPbObI4rVc_QDp9i5tD

Description
This project involves developing a Virtual Hospital (VH) and Synthetic Patient (SP) capability to simulate realistic electronic health records including categorical, discrete, continuous, imaging, text and biomedical specimen data. This relates to the unstructured-DataSifter, synthetic text generation, text-obfuscation, and text-mining/inference.

Student Skills
● EECS, Stats/Biostats/Math, Computer Science (CSE/CS-LSA) and School of Information (SI)
● UI/UX design, HTML5, JavaScript

Project Goals
● Go through the Training Modules, practice HTML/JS/Angular/Node programming
● Get your GitHub domain going
● Choose 1-2 deliverables, go over current design, start expansion, include unit tests, pilot development
● Coordinate with team

Deliverables
● Learn the current state of the project (Simeone)
● Examine the prior datasets (NHANES, MIMIC, etc.)
● Choose categorical, discrete, continuous, imaging, text and biomedical specimen data and illustrate examples
● Experiment and demo 1,000 VH/SP cases
● Validate VH/SP using machine learning classification, prediction, clustering and analytics modules.

Team Activities
● Weekly team meetings
● Code review (pull/push Github requests)
● Join/present the SOCR All-Hands Calls (twice a month, SOCR BlueJeans channel)

References
● Synthetic data generation using Variation Auto-encoders/auto-decoders.

● We can explore data augmentation and synthetic generation variational AE-AD neural network to generate synthetic cases (using the decoder process, i.e., predict function of the NN model) and explore the similarities between the joint distributions of the original data and the synth-data.


● https://towardsdatascience.com/auto-encoder-what-is-it-and-what-is-it-used-for-part-1-3e5c6f017726


● SOCR example https://socr.umich.edu/HTML5/ABIDE_Autoencoder/

https://www.socr.umich.edu/html/SOCR_Research.html
● **Charlatan R package:** [https://cran.r-project.org/web/packages/charlatan/index.html](https://cran.r-project.org/web/packages/charlatan/index.html)


● **stringdist** package allows us to compare strings/text: [https://cran.r-project.org/web/packages/stringdist](https://cran.r-project.org/web/packages/stringdist)

  ○ Also see: [https://becominghuman.ai/generative-adversarial-networks-for-text-generation-part-1-2b886c8cab10](https://becominghuman.ai/generative-adversarial-networks-for-text-generation-part-1-2b886c8cab10)


● **Autoencoder approach:** [https://github.com/stas-semeniuta/textvae](https://github.com/stas-semeniuta/textvae) and [https://www.aclweb.org/anthology/D17-1066.pdf](https://www.aclweb.org/anthology/D17-1066.pdf)
I. Kime_CircleCloud_WebApp_index.html
This is a pure JavaScript/HTML5 app that needs some bug-fixing, improvements, and feature enhancements to show the dynamics of the natural attraction-repelling forces.

II. SOCR_Vase_TCIU_Model.html
This is an R-source (Rmd) that is knitted/exported as pure HTML5/JS.

III. SOCR_BivariateNormalDistribution Webapp

How to expand the Bivariate-Normal to any Bivariate Distribution? Define the 2 marginal distributions and use this protocol to specify their association (in terms of the correlation) to derive the joint Bivariate distribution PDF:

This is an R-source (Rmd) that is knitted/exported as pure HTML5/JS.

Below is a starting JavaScript implementing the basic Bivariate Normal Distribution functions.

We want to expand the current app (see it live here and the code (BVN.zip) is here), to include in addition to the MCMC simulation-based (approximate) calculations, which is already implemented, ADD an exact PDF/CDF based calculations. Include a new check-box to allow the user to specify exact vs. approximate calculation (just like we have for with/without using WebGL for the visualization.

```javascript
function normalCDF(X)
    // using Hastings algorithm with maximal error=10^{-6}
    var T=1/(1+.2316419*Math.abs(X));
    var D=0.3989423*Math.exp(-X*X/2);
    var Prob=D*T*(.3193815+T*(-0.3565638+T*(1.781478+
        T*(-1.821256+T*1.330274))));
    if (X>0) {Prob=1-Prob}
    return Prob
}

function binormalCDF(x,y,R){    // P(X>x,Y>y;R)
    with (Math){
        var s=(1-normalCDF(x))*(1-normalCDF(y));
        var sqr2pi=sqrt(2*PI);
    }
```
var h0=exp(-x*x/2)/sqr2pi;
var k0=exp(-y*y/2)/sqr2pi;
var h1=-x*h0;
var k1=-y*k0;
var factor=R*R/2;
s=s+R*h0*k0+factor*h1*k1;
var n=2;
while ((n*(1-abs(R))<5) && (n<101)) {
    factor=factor*R/(n+1);
    h2=-x*h1-(n-1)*h0;
    k2=-y*k1-(n-1)*k0;
    s=s+factor*h2*k2;
    h0=h1;
    k0=k1;
    h1=h2;
    k1=k2;
    n=n+1;
}
var v=0;
if (R>.95) {
    v=1-normalCDF(max(h,k))
    s=v+20*(s-v)*(1-R);
} else if (R<-.95) {
    v=abs(normalCDF(h)-normalCDF(k))
    s=v+20*(s-v)*(1+R);
}
}
return s;

function BVN( ) {
    // The following user inputs are necessary
    X, Y, M1, M2 (means), and S1, S2 (sigma1 and sigma 2),
    and Rho (correlation)
    Prob="NaN";
    if ((S1<=0) || (S2<=0)){
        alert("The standard deviations must be positive.");
    } else if ((R<-1) || (R>1)){
        alert("The correlation coefficient must be between -1 and +1.");
    } else {
        h=-(X-M1)/S1;
        k=-(Y-M2)/S2;
        Prob=binormalCDF(h,k,R);
        Prob=Math.round(100000*Prob)/100000;
    }
    return(Prob);
}
SOCR 2021 MDP Project: Java Applets Code → HTML/JS Apps

SOCR Project Leaders: Ivo Dinov
Website: various
GitHub: https://github.com/SOCR
          https://github.com/SOCR/SOCR-Java

Training Modules: https://github.com/SOCR/socr-tutorials
GDrive: TBD

Convert some of the old SOCR Java Applets to modern HTML5/JavaScript apps

Pilot Project: Start with the SOCR Fourier/Wavelet Game applet

SOCR 1D Fourier / Wavelet signal decomposition into magnitudes and phases (Java applet)

Top-panel: original signal (image), white-color curve drawn manually by the user and the reconstructed synthesized (IFT) signal, red-color curve, computed using the user modified magnitudes and phases
Bottom-panels: the Fourier analyzed signal (FT) with its magnitudes and phases
http://www.socr.ucla.edu/htmls/game/Fourier_Game.html (Java Applet - Run in Java-enabled IE)


Additional critical Java Apps to convert to HTML5/JS:
- LLN:
  - http://socr.ucla.edu/htmls/exp/LLN_Simple_Experiment.html
- CLT
  - http://socr.ucla.edu/htmls/exp/Sampling_Distribution_CLT_Experiment.html
- π and e stochastic estimation experiments:
  - http://socr.ucla.edu/htmls/exp/Uniform_E-Estimate_Experiment.html
- Polynomial model fitting:
  - http://socr.ucla.edu/htmls/exp/Uniform_E-Estimate_Experiment.html
SOCR 2021 MDP Project: **Topological Data Analysis (TDA), Persistent Homology, and Betti Numbers for Point-cloud Data**

**SOCR Project Leaders:** Ivo Dinov  
**Website:** TBD  
**GitHub:** TBD

**Training Modules:** see background papers, R-packages, and tutorials listed below  
**GDrive:** https://drive.google.com/drive/folders/1QZLjH_xN_SSsrjF2jsc9mDQBUCdDKPIY

**Description:**  
This project aims to expand the SOCR lab data analytical capabilities using advanced topological representations. Data are viewed as high-dimensional point clouds. These are interpreted as samples through a high-dimensional manifold which will be modeled using simplicial complexes. Using the simplicial decomposition, we can compute persistent homology, Betti numbers, and derive other topological metrics that can be used for classification, regression and clustering.

**References:**

- **Papers**

- **Tutorials**
  - See DSPA Chapter 5: [http://www.socr.umich.edu/people/dinov/courses/DSPA_notes/05_DimensionalityReduction.html](http://www.socr.umich.edu/people/dinov/courses/DSPA_notes/05_DimensionalityReduction.html)

- **R-Code**
  - [https://cran.r-project.org/web/packages/TDA/TDA.pdf](https://cran.r-project.org/web/packages/TDA/TDA.pdf)

- **Test data:** [https://umich.instructure.com/courses/38100/files/folder/Case_Studies](https://umich.instructure.com/courses/38100/files/folder/Case_Studies)