**By:** Thanos Gentimis

**Course:** Digital Agriculture

Contents

[**I. R 1**](#_heading=h.gjdgxs)

[1. Getting started with R 1](#_heading=h.30j0zll)

[2. R basic links 1](#_heading=h.1fob9te)

[3. Reading files into R 1](#_heading=h.3znysh7)

[4. R trinkets 1](#_heading=h.2et92p0)

[5.](#_heading=h.tyjcwt) An amazing intro to R

[6. Daily task 01/25/2022 2](#_heading=h.3dy6vkm)

[7. Daily Task 01/31/2022 2](#_heading=h.1t3h5sf)

[8. Advanced Reads in R 2](#_heading=h.4d34og8)

[9. Links recommended by LSU library 3](#_heading=h.2s8eyo1)

[**II. TIME SERIES ANALYSIS 3**](#_heading=h.17dp8vu)

[1. Notes 3](#_heading=h.3rdcrjn)

[2. Information on Time Series Analysis 3](#_heading=h.26in1rg)

[3. Daily Task 02/01/2022 3](#_heading=h.lnxbz9)

[4. Daily Task 02/07/2022 4](#_heading=h.35nkun2)

[5. Daily Task 02/08/2022 4](#_heading=h.1ksv4uv)

[6. Time Series Analysis on Agriculture 4](#_heading=h.44sinio)

[7. Various concepts related to Time Series 5](#_heading=h.2jxsxqh)

[**III. PYTHON 5**](#_heading=h.z337ya)

[1. Notes 5](#_heading=h.3j2qqm3)

[2. Getting started 5](#_heading=h.1y810tw)

[3. Scripting in python 5](#_heading=h.4i7ojhp)

[4. Daily Task 02/14/2021 5](#_heading=h.2xcytpi)

[**IV. MACHINE LEARNING: NEURONAL NETWORK 7**](#_heading=h.1ci93xb)

[1. Notes and presentations 7](#_heading=h.3whwml4)

[2. What is a Neural Network 7](#_heading=h.2bn6wsx)

[3. Daily Task 02/15/2022 7](#_heading=h.qsh70q)

[4. Essentials for Convolutional Neural Networks 8](#_heading=h.3as4poj)

[5. Daily Task 02/21/2022 8](#_heading=h.1pxezwc)

[6. Daily Task 03/07/2022 9](#_heading=h.49x2ik5)

[7. Higher Level Links on Machine Learning and Fun reads 9](#_heading=h.2p2csry)

[**V. MACHINE LEARNING: RANDOM FOREST 10**](#_heading=h.147n2zr)

[1. Notes 10](#_heading=h.3o7alnk)

[2. Daily Task 03/22/2022 10](#_heading=h.23ckvvd)

[3. Daily Task 03/23/2022 10](#_heading=h.ihv636)

[4. Various links for Random Forests 10](#_heading=h.32hioqz)

[5. Daily task 03/28/2022 11](#_heading=h.1hmsyys)

[**VI. MACHINE LEARNING: SUPPORT VECTOR MACHINES 11**](#_heading=h.41mghml)

[1. Notes 11](#_heading=h.2grqrue)

[2. Essential Guides to Coding an SVM and definitions 11](#_heading=h.vx1227)

[3. Daily Task 03/07/2022 11](#_heading=h.3fwokq0)

[4. Daily Task 03/08/2022 12](#_heading=h.1v1yuxt)

[**VII. QGIS 12**](#_heading=h.4f1mdlm)

[1. QGIS basic setup 12](#_heading=h.2u6wntf)

[2. QGIS video tutorials 13](#_heading=h.19c6y18)

[3. Daily Task 03/29/2022 13](#_heading=h.3tbugp1)

[4. Shapefile Databases 13](#_heading=h.28h4qwu)

[**VIII. ArcGIS 14**](#_heading=h.nmf14n)

[1. ArcGIS Starting Kit 14](#_heading=h.37m2jsg)

[2. Introduction to ArcGIS 14](#_heading=h.1mrcu09)

[3. Basic commands in ArcGis 14](#_heading=h.46r0co2)

[4. Daily Task 2/19/2019 14](#_heading=h.2lwamvv)

[5. Daily Task 2/25/2019 15](#_heading=h.111kx3o)

[**IX. IMAGE PROCESSING 16**](#_heading=h.3l18frh)

[1. Imager- an R package for image processing 16](#_heading=h.206ipza)

[2. Daily Task 04/04/2022 16](#_heading=h.4k668n3)

[**X. IMAGES IN PYTHON 16**](#_heading=h.2zbgiuw)

[1. Basic Libraries and their links 16](#_heading=h.1egqt2p)

[2. Daily Task 04/11/2022 17](#_heading=h.3ygebqi)

[**XI. IMAGE STITCHING IN PYTHON 18**](#_heading=h.2dlolyb)

[1. Image stitching 18](#_heading=h.sqyw64)

[2. Daily Task 04/12/2022 18](#_heading=h.3cqmetx)

[**XII. DATABASES, POSTGRESQL 19**](#_heading=h.1rvwp1q)

[1. PostgreSQL-Installation and intro 19](#_heading=h.4bvk7pj)

[2. PostgreSQL tutorials 19](#_heading=h.2r0uhxc)

[3. Daily Task 04/18/2022 19](#_heading=h.1664s55)

[**XIII. DATA IN AGRICULTURE 20**](#_heading=h.3q5sasy)

[1. Available databases 20](#_heading=h.25b2l0r)

[2. Daily Task 04/25/2022 21](#_heading=h.kgcv8k)

[**XIV. USEFUL LINKS-PRESENTATIONS AND VARIOUS OTHERS 22**](#_heading=h.34g0dwd)

[1. Digital Ag Day Spring 2019 22](#_heading=h.1jlao46)

[2. API's 22](#_heading=h.43ky6rz)

# R

## Getting started with R

One of the biggest hurdles with any new software/language is how to get it started. The following links will get you on your way.

* Download R from here: (any mirror will do): <https://www.r-project.org/>
* Download R-Studio from here (the free version): <https://www.rstudio.com/>
* What is R?:<https://www.r-project.org/about.html>
* R studio introduction: <https://www.youtube.com/watch?v=jPk6-3prknk>

## R basic links

* [The R Cookbook](http://www.cookbook-r.com/)
* [R intro](https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf)
* [The Pirate's Guide to R](https://drive.google.com/file/d/0B4udF24Yxab0S1hnZlBBTmgzM3M/view)
* [cran.r-project.org](https://cran.r-project.org/)
* [R-studio](https://www.rstudio.com/)
* [R for Beginners](https://cran.r-project.org/doc/contrib/Paradis-rdebuts_en.pdf)
* [Basic R Examples](http://www.rexamples.com/)
* [A great Introductory Course to Data Analysis with R](https://moderndive.com/index.html)\

## Reading files into R

One of the hardest aspects in data science is how to read the data file in the program of your choice.

Fear not. The folks on DataCamp have done an amazing job collecting the various ways to insert various types of files in R.

[Follow this link](https://www.datacamp.com/community/tutorials/r-data-import-tutorial)

## R trinkets

Over the years I have been creating a bunch of small R codes to use in teaching and research.

OBVIOUSLY, I never memorize the codes, so in order NOT to re-discover them every time I have created a repository on google drive called R-trinkets, in which I put self-containing pieces of code, with the corresponding datasets and result files.

That combined with the comments I put in the codes, allows me to quickly adapt the codes into other situations,

The following link will allow you to view and download the files (at some point I might increase the access privilege to allow you to change things but for now ... I am guarding my precious pieces of code!)

<https://drive.google.com/drive/folders/1UUJL8qIX0OEL-si9QMNZA4Ft6PcQoFmq?usp=sharing>

## An amazing intro to R written by Frank Saforo

<https://moodle.lsu.edu/pluginfile.php/1344141/mod_resource/content/1/lab01_practice.html>

## Daily task 01/25/2022

Follow this link and let’s analyze together how we can read and merge multiple files with the same format using R

<https://drive.google.com/drive/folders/1Yovow4AI5AyUNe6XfZ-O-Sf4MZ1PyTsK?usp=sharing>

Then, let’s find out who did homework 1 right by following this link

<https://drive.google.com/drive/folders/1Wn25TOJzYjiPoaykNngHgPEawQeXAjJh?usp=sharing>

## Daily Task 01/31/2022

In today’s task we will follow a simple filtering and data exploration code. We will learn about custom functions, summary statistics, and simple analysis with some basic graphs. Follow the link below and make a clone of the repository to your computer. Make sure you keep the names and the structure the same

<https://drive.google.com/drive/folders/1KzOtxux2yWHh7TNkX6qC888LQ4POzTMC?usp=sharing>

We will be working with the code TG\_RExploration\_1\_1 . Make sure you create a copy of it in your own repository and use your initials. Put as many comments as possible as well as try to expand the code with your own ideas.

## Advanced Reads in R

Dear all,

I will add here various interesting links I have found related to R. This is mostly a repository for things I find cool and useful, but feel free to contribute.

* [How to avoid pitfalls in graphing](https://www.r-bloggers.com/data-visualisation-pitfalls-how-to-avoid-barbarplots/)
* [Sub-setting huge data-sets](https://www.r-bloggers.com/playing-map-and-reduce-in-r-subsetting/)
* [Hot to create sub-strings in R](https://www.sharpsightlabs.com/blog/substring-in-r/)
* [Machine Learning Results In R](https://www.r-bloggers.com/machine-learning-results-in-r-one-plot-to-rule-them-all/)
* [Creating stylish tables](https://www.littlemissdata.com/blog/prettytables)
* [Introduction to dplyr](https://cran.r-project.org/web/packages/dplyr/vignettes/dplyr.html)
* [Reshaping your data with tidyr](https://uc-r.github.io/tidyr)
* [Writing books with R Markdown](https://bookdown.org/yihui/bookdown/figures.html)

## Links recommended by LSU library

* The R Book: <https://onlinelibrary.wiley.com/doi/book/10.1002/9781118448908>
* Understanding Statistics Using R: <https://link.springer.com/book/10.1007%2F978-1-4614-6227-9>
* Introduction to statistics and data analysis: with exercises, solutions and applications in R: <https://link.springer.com/book/10.1007%2F978-3-319-46162-5>
* A Course in Statistics with R: <https://onlinelibrary.wiley.com/doi/book/10.1002/9781119152743>

# TIME SERIES ANALYSIS

In this section we will explore an overlooked type of data that is very prominent in Agriculture and more. I am talking of course about time-series which are nothing more than data indexed by some sort of time variable.

## Notes

Here are some notes/presentations on Time Series

* The Introduction to Timeseries contains some theoretical underpinnings of timeseries and it is due to InKwan Yu, an educator at the University of Illinois.
* The Time Series Forecasting is a presentation wrapping up this section.

\*You can find the ppts into the folder Notes\_1

## Information on Time Series Analysis

* [A little book of R for Time Series](https://a-little-book-of-r-for-time-series.readthedocs.io/en/latest/)
* [What is a Time Series the wiki version](https://en.wikipedia.org/wiki/Time_series)
* [An introduction to time series characteristics](https://onlinecourses.science.psu.edu/stat510/node/47/)

## Daily Task 02/01/2022

Today's task will help us understand a few things about manipulating and presenting time series:

Navigate to this google drive folder:

<https://drive.google.com/drive/folders/1HYskI8SBbjnVR3wlqboH_r6BhJfKHTd1?usp=sharing>

We will be analyzing the file TG\_TimeSeries\_1\_2.R and learning the basics for time series analysis in Weather data.

Please add comments and extend the code.

Finally, if we have time, we will review the information in this webpage:

<https://anomaly.io/seasonal-trend-decomposition-in-r/>

## Daily Task 02/07/2022

**A)** In today's daily task we will start by downloading information through R automatically using an API

* First download everything in this google repository: <https://drive.google.com/drive/folders/1GlVcl6Dwm-mMFj04PlK6qoOMNy8nHGjW?usp=sharing>
* Then navigate to the page of Quandl: <https://www.quandl.com/>
* Create a free account using your lsu id and mark the fact that you are a student. Navigate to account settings and copy paste your API key.
* Let's open up the TG\_API\_1\_1.R in the folder codes. Make sure you put as many comments as needed and follow along!

**B)** Let's try a different way of downloading information, this time from Yahoo finance. No need for an account here! Let's open up the file TG\_Forecasting\_1\_1.R

* We will first download various timeseries tables using the package tidyquant and format them in a big table.
* Then we will try to see if various timeseries are correlated.
* Finally, we will do short term predictions using various predictive models and compare their accuracy.

## Daily Task 02/08/2022

In today’s daily task we will continue working on short term prediction with time series, this time using the Covid 19 virus data.

* Navigate to this folder and make sure you download everything. Once again, the structure matters:

<https://drive.google.com/drive/folders/1JAkzTe3l6mDbwZVPhQl4NCMM9WahLkew?usp=sharing>

* We will be working with the code TG\_TimeSeriesAnalysis\_1\_2
* Make sure you make a copy of it using your own initials and follow along making comments as you go.
* In the folder you can also find the TG\_CodeDescription\_1\_1.Rmd which is an R markdown code. This helps create pdf and html documents that have "active" R codes and are heavy in mathematical notation. If I complete it on time, we can analyze that as well.

## Time Series Analysis on Agriculture

In this page I will list some fun reads for the use of time series analysis in agriculture plus

some datasets that you might find useful.

* [Time series analysis of soybean response to varying atmospheric conditions for precision agriculture](https://www.semanticscholar.org/paper/Time-series-analysis-of-soybean-response-to-varying-Larbi-Green/e61a4b96981dfb87516641aae26aac67c67e2d05)
* [Special Issue "Precision Agriculture and Remote Sensing Data Fusion"](https://www.mdpi.com/journal/sensors/special_issues/PA_remote)
* [Time Series Data Library](https://datamarket.com/data/list/?q=cat%3Ag22+provider%3Atsdl)
* [Connecting to a Weather API](https://cran.r-project.org/web/packages/rwunderground/README.html)

## Various concepts related to Time Series

Here I will post links that help explain the theory behind time series analysis.

* [Time Series Forecasting](http://r-statistics.co/Time-Series-Forecasting-With-R.html)
* [Forecasting Principles and Practices](https://otexts.org/fpp2/)
* [Quick R Time Series Analysis](https://www.statmethods.net/advstats/timeseries.html)
* [Seasonality and Trend Analysis](https://anomaly.io/seasonal-trend-decomposition-in-r/)
* [Neural Networks for Forecasting](https://www.packtpub.com/mapt/book/big_data_and_business_intelligence/9781783989683/9/ch09lvl1sec67/neural-networks-for-forecasting)
* [Dygraphs for R](https://rstudio.github.io/dygraphs/)
* [Quandl](https://www.quandl.com/)
* [Root Mean Square Error in R](https://www.geeksforgeeks.org/root-mean-square-error-in-r-programming/)
* [Timeseries Forecasting Using Python](https://machinelearningmastery.com/time-series-forecasting-methods-in-python-cheat-sheet/)

# PYTHON

In this section I will share all information about our second favorite programming language Python!

## Notes

\*You can find the ppts in the folder Notes\_2

## Getting started

In this page you will find a collection of useful links for python, including how to install and start it. You can also find free books and some tutorials.

* [Python the official website](https://www.python.org/)
* [ANACONDA a collection of python interpreters, packages and basic UI's](https://www.anaconda.com/download/)
* [Automate the boring stuff with Python](https://automatetheboringstuff.com/)
* [Think Stats with Python](http://greenteapress.com/thinkstats2/html/index.html)
* [Think Python](http://www.greenteapress.com/thinkpython/html/index.html)

## Scripting in python

Here you can find some simple codes in python to get you started.

You can use the discussions at the end of this section to add your own scripts or useful links.

* [Official basic simple programs](https://wiki.python.org/moin/SimplePrograms)
* [Basic commands for dataset handling in the pandas library](https://towardsdatascience.com/23-great-pandas-codes-for-data-scientists-cca5ed9d8a38)
* [Programming ArcGis with python](http://libezp.lib.lsu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edsknv&AN=edsknv.kpPAGISP02&site=eds-live&scope=site&profile=eds-main)
* [Learn Enough python to work some simple codes](http://libezp.lib.lsu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=cat00252a&AN=lalu.4596230&site=eds-live&scope=site&profile=eds-main)
* [Advanced scripting in python](https://github.com/Avik-Jain/project-based-learning#python)
* [100 days of Machine Learning with python](https://github.com/Avik-Jain/100-Days-Of-ML-Code)

## Daily Task 02/14/2021

In today's daily task we will explore the most basic linear regression algorithm with a Python code.

**A) Getting the data and code.**

* Navigate to the the repository: <https://drive.google.com/drive/folders/1pEYWN9M1GP2_9oPeGYko2CSSU66jGJno?usp=sharing>
* Download everything keeping the structure.
* Make sure you save the folder in the same root directory as anaconda. So, if you installed anaconda on the c drive, the folder with the linear regression needs to be on the C drive as well. Later on, this will not be an issue, but if you want to use jupyter notebook you have to do this.
* Make a copy of the code TG\_LinearRegression\_2\_1 and use your initials.

**B) Setting up the environment.**

* Go to the environments tab and select the environment that you have created for this course. If you don't have one create using the plus button in the bottom and call it Digital\_Ag, or whatever else you want.
* We will be installing a bunch of helpful packages. This process needs to happen once, but packages should be updated periodically.
* Go to the tab Install and change it to "Not installed". Then in the search package field type:

1. scikit-learn . Select the package so that a green down arrow appears in the box and hit apply (bottom right corner). The program will ask you to install some dependencies. Click apply and wait.
2. pandas .  Select the package so that a green down arrow appears in the box and hit apply (bottom right corner). The program will ask you to install some dependencies. Click apply and wait.
3. matplotlib .  Select the package so that a green down arrow appears in the box and hit apply (bottom right corner). The program will ask you to install some dependencies. Click apply and wait.
4. seaborn .  Select the package so that a green down arrow appears in the box and hit apply (bottom right corner). The program will ask you to install some dependencies. Click apply and wait.

* When you are done with the packages, return to the home tab and make sure you "install" jupyter notebook. You are basically connecting the jupyter notebook to this environment, you are not installing it from scratch.

**C) In the Jupyter notebook**

* Launch the Jupyter Notebook application and navigate through the browser to the place you have saved the folder.
* Run the TG\_LinearRegression\_2\_1. Make sure you follow along and add comments.
* There are two ways to add comments in the notebook.

1. One is to use the "#" (tab delimiter). Python knows not to interpret anything that is on the same line as a tab
2. You can add a new cell and change its designation from code to markdown. This is useful for larger comments. Also, this is a way to create documents that combine text with working code (VERY USEFULL!)

# MACHINE LEARNING: NEURONAL NETWORK

## Notes and presentations

\*You can find the ppts in the folder Notes\_3

## What is a Neural Network

* By far the easiest intro to Neural Networks I know: <https://deeplizard.com/learn/video/hfK_dvC-avg>
* Neural Networks in a minute: <https://www.youtube.com/watch?v=rEDzUT3ymw4&t=1s>
* How a Neural Network Learns: <https://deeplizard.com/learn/video/_N5kpSMDf4o>

## Daily Task 02/15/2022

**A) Finishing the homework**

* Navigate again to the folder for the linear regression:

<https://drive.google.com/drive/folders/1pEYWN9M1GP2_9oPeGYko2CSSU66jGJno?usp=sharing>

* This time let's work on the TG\_LinearRegression\_3\_1 which you can find under the folder coders.
* Make sure you are in the appropriate environment for Digital Agriculture from the anaconda navigator. We will again use jupyter notebook and make sure the file structure is identical.
* Follow along and make comments!

**B) Installing Packages for Neural Networks**

* In the Digital Ag Environment go to the tab Not Installed and type in Keras. This is one of the most prominent Machine Learning packages in Python. Install it including all the dependencies.
* Also check to see if you have the package tkinter. If not, add that as well.

**C) Installing Spider**

* Spyder is another python editor, like jupyter notebook, but the philosophy there is geared towards coding rather than presentation.
* Make sure you are in the digital ag Environment and click install at the Spyder app
* We will customize Spyder to look very similar to RStudio!

**D) Getting the codes and data**

* Navigate to the folder:

<https://drive.google.com/drive/folders/1od7lI0oD8xd426Z3BgsJOdgY8KXlEzJE?usp=sharing>

* Download everything, keeping the structure. You can save this folder **ANYWHERE** in your computer.
* We will be working with the code TG\_NN\_1\_1
* Open up Spyder and navigate to the folder using the file tab on the top left.
* You can change Spyder's layout through the tab view, window layout
* Also, from tools/preferences/syntax coloring, you can switch the theme to anything you like (and why is that not spyder dark???)

**E) Reviewing the code**

* We will go over the code line by line and explain what it does. You can again put in comments using the # delimiter.
* This will take 2 classes so we will take it slow.

## Essentials for Convolutional Neural Networks

Here you can find a few links that dig a bit deeper in what we discussed today. Read them at your own discretion:

a) A Step by step code explanation of the arguments in the code we used in class can be found here:

<https://becominghuman.ai/building-an-image-classifier-using-deep-learning-in-python-totally-from-a-beginners-perspective-be8dbaf22dd8>

b) The wikipedia page with some generalities on the subject:

<https://en.wikipedia.org/wiki/Convolutional_neural_network>

c) The library Keras we used:

<https://keras.io/>

d) A simple intro to Neural Networks

<https://www.youtube.com/watch?v=JBlm4wnjNMY&list=PL8dPuuaLjXtNM_Y-bUAhblSAdWRnmBUcr&index=42>

e) An intro to Convolutional Neural Networks:

<https://www.youtube.com/watch?v=YRhxdVk_sIs>

This channel is a great high level but also technical resource for machine learning and more. For example this list is fantastic:

<https://www.youtube.com/watch?v=gZmobeGL0Yg&list=PLZbbT5o_s2xq7LwI2y8_QtvuXZedL6tQU&t=3s>

## Daily Task 02/21/2022

Today we will try to understand how neural networks can be used for image analysis.

**A) Intro to the pre-trained image classifier called YOLO (I know I know ...) or darknet YOLO**

<https://pjreddie.com/darknet/yolo/> There are various implementations of it and we will use the one for R.

* Please open the following folder:

<https://drive.google.com/drive/folders/1cn21lb5MaHpoIjxkp_UBzkxH7G2gmFqm?usp=sharing>

* Download everything and save it in the folder of your choice.

**B) Setting up the packages for the code. Make sure you put comments.**

* Open up the TG\_darknetclassifier\_1\_1.R code.
* Run the first two lines of code to install the package dlib and the package darknet.
* If you don't have devtools, you will be prompted to install the Rtools from R studio. Agree, and go get some water. This might take a while.
* After devtools are there, rerun the second line making sure that this time it was installed.
* If you still get an error, un-comment lines 3 to 5 and run them. This should fix it. If not, sorry we have reached a dead end, just follow along.

**C)** **Running the code.  Make sure you put comments.**

* Change the working directory to the folder of your choice (Session ... Change working directory etc etc), place the image you want to classify on the folder of your choice and copy down the path (you will need the full path I am afraid).
* Copy that path in line 18. You are instructing yolo to go find the image there. **MAKE SURE YOU MANUALLY CHANGE ALL FORWARD SLASHES TO BACKWARD ONES**... annoying I know.
* That is it. Select lines 13 to 20 and run the code. You should now have an image in that folder called "prediction" marvel at the results (well sort of)

## Daily Task 03/07/2022

We are now ready to write our first Convolutional Neural Network Classifier

1) Open up Anaconda Navigator and start a new python 3 code.

2) Follow the link here to the actual code:

<https://drive.google.com/drive/folders/1qm_Lv6ukacw7Rtyf-qRsA3i6ZXAw9ieQ?usp=sharing>

3) We will be running the code piece by piece explaining what it does in the process.

## Higher Level Links on Machine Learning and Fun reads

* [Introduction to Deep Learning](https://www.datasciencecentral.com/profiles/blogs/introduction-to-deep-learning)
* [Linear regression explained](https://towardsdatascience.com/linear-regression-part-1-types-examples-gradient-descent-example-2e8c22b05f61)
* [Image classification workflow](https://towardsdatascience.com/image-classifier-complete-workflow-a9e0003c80ba)
* [Train test split and cross validation in Python](https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6)
* [Chicago crime mapping](https://hackernoon.com/chicago-crime-mapping-magic-of-data-science-and-python-f2ecad74a597)
* [What on earth is Data Science](https://hackernoon.com/what-on-earth-is-data-science-eb1237d8cb37)
* [Process Improving Using Data](https://learnche.org/pid/contents) (Free Book)
* [Convolutional Neural Networks for Beginners](https://na01.safelinks.protection.outlook.com/?url=https%3A%2F%2Ftowardsdatascience.com%2Fconvolutional-neural-networks-for-beginners-practical-guide-with-python-and-keras-dc688ea90dca&data=02%7C01%7Ctgentimis%40agcenter.lsu.edu%7C014fea4715e240adc55408d622a2ca9f%7C804b509899084bdf9c06b3df777563aa%7C0%7C0%7C636734479917502828&sdata=WamWbClYdraoRkBXxfH%2FDaH36a8Fa3ojARVbxXa9nmQ%3D&reserved=0)
* [A complete Guide to KNN](https://kevinzakka.github.io/2016/07/13/k-nearest-neighbor/)
* [Change the ways you interact with Technology](https://hbr.org/2017/01/deep-learning-will-radically-change-the-ways-we-interact-with-technology)
* [High Performance Computing at LSU](http://www.hpc.lsu.edu/training/tutorials.php)
* [Main Types of Neural Networks](https://medium.com/towards-artificial-intelligence/main-types-of-neural-networks-and-its-applications-tutorial-734480d7ec8e)
* [A guide to Deep Learning and Neural Networks](https://serokell.io/blog/deep-learning-and-neural-network-guide)
* [Feature selection the right way](https://towardsdatascience.com/learn-how-to-do-feature-selection-the-right-way-61bca8557bef)

# MACHINE LEARNING: RANDOM FOREST

## Notes

\*You can find the ppts in the field Notes\_5

## Daily Task 03/22/2022

**A) Understanding Decision Trees**

* First let's review the draft from Alex Tryphoros' paper to get an idea of what a Decision tree and a Random Forest is. This can be found in the Notes and Presentations section.
* Then download everything you find in the following link:

<https://drive.google.com/drive/folders/1UnFPK1OPX2GNX8OUUYg5EzCKE-m9ppFt?usp=sharing>

* For today we will start with the python code TG\_TreeVisualizing\_1\_1 which you can find in the python folder of the repository above. Follow along adding comments when needed.

**B) A link to Dr' Li's Class**

* One of the priorities of this class is to introduce you to various Machine Learning tools amongst other things.
* If you are interested in going deeper with these concepts there are multiple classes from the CS and EXST that you can take.
* One of them is Dr. Bin Li's class, which I will be using to introduce Random Forests.

Please find here: <http://statweb.lsu.edu/faculty/li/teach/exst7087/> the material for today's class.

**C) Working on the code**

* We will work on an updated version of Dr. Li's code that you can find in the R folder of the repository I gave you.
* The code is TG\_RF\_1\_1.R
* Download it and save it somewhere. The code is using internal data so there is really no need for structure preservation.
* Follow along and comment along the way.

## Daily Task 03/23/2022

* Download everything from the following folder keeping the structure:

<https://drive.google.com/drive/folders/1UnFPK1OPX2GNX8OUUYg5EzCKE-m9ppFt?usp=sharing>

* This is an example of using Random Forests to do numeric predictions (regression) not just classification.
* We will work with the code TG\_RF\_2\_2 in Spyder
* Make sure you put as many comments as you can and follow along.

## Various links for Random Forests

Here you can find various links if you want to expand your knowledge of random forests:

* [The Random Forest Algorithm – Towards Data Science](https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd)
* [Random Forest Simple Explanation](https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d)
* [The wiki page](https://en.wikipedia.org/wiki/Random_forest)
* [More about Random Forests](https://builtin.com/data-science/random-forest-algorithm)

## Daily task 03/28/2022

**A) Getting the data:**

* Download everything from the following link making sure you keep the same structure.

<https://drive.google.com/drive/folders/1puWpCeFzMELGU5VK2gVNSE_2L-tTVD_a?usp=sharing>

* We will be working with the code TG\_Comparison\_1\_1. Make sure you rename it using your initials.
* This code compares various machine learning models. Make sure you put as many comments as needed so that you completely understand what is going on.
* As soon as we all have the code in front of us up and running, I will get you to work in 2 groups with 3 to 4 people. Your task is:
* Explain what each section of the code does. Make sure that everybody understands.
* Increase the number of repetitions to 10. Report the results.
* You will have 1 hour to work on that.

**B) Review it with me.**

* We will go over it together at the end of the hour.
* Complete your notes and ask questions.

# MACHINE LEARNING: SUPPORT VECTOR MACHINES

In this section we will discuss the second most prominent tool in Machine Learning, i.e., Support Vector Machines (SVM). As a mathematician this is probably my favorite topic, so be prepared for absolute nerd out!

## Notes

\*You can find the ppts in the folder Notes\_4

## Essential Guides to Coding an SVM and definitions

* [An illustrated example and explanations of a SVM](https://medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72)
* [SVM's with examples of tunning](https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/)
* [The wikipedia page](https://en.wikipedia.org/wiki/Support_vector_machine)
* [A fun to watch youtube intro](https://www.youtube.com/watch?v=Y6RRHw9uN9o)

## Daily Task 03/07/2022

In today’s Daily task we will do some theoretical Introduction to SVM's and review one example:

**A) Intro to SVM's**

* Print the notes you will find on this section about SVM's
* Fill in the empty parts, ask questions and put comments

**B) Following a blog-example**

* Open up your anaconda navigator, go to the digital ag environment and create a new file called Initials\_SVM\_1\_1
* Navigate to this page:

<https://jakevdp.github.io/PythonDataScienceHandbook/05.07-support-vector-machines.html>

* This is a great example to understand the theory and also have a working algorithm. Follow along pasting in cells one by one.
* Add as many comments as needed.

## Daily Task 03/08/2022

Today we will continue SVM's and follow a couple simpler examples.

* Please navigate to this folder and download everything to your machine.

<https://drive.google.com/drive/folders/1qL7AMsv-DFV5bqiYAZ2NIa5lvQPyXP9O?usp=sharing>

* Make sure you clone everything, and you are using the appropriate environment in Anaconda.
* We will be working on the TG\_SVM\_2\_1 python script. Make sure you create a version using your own initials.
* This SVM will try to identify fraudulent checks by using some measures on the signatures.
* Your task is to put comments in the code, explaining what it does.
* This week's homework is connected to this script, since you will be replicating it with your own data.

# QGIS

QGIS is a free analog to ArcGIS the standard software for mapping geolocated data. QGIS has almost the same capabilities as ArcGIS, has a vibrant community of supporters, integrates with other software and programming languages and is completely free. Other classes will go in depth here, but we will just scratch the surface and make sure you can create a few maps to display outputs.

## QGIS basic setup

Dear all,

a) Download QGIS for your platform: QGIS is available on Windows, MacOS X, Linux and Android"

<https://www.qgis.org/en/site/forusers/download.html>

b) Review this Series of tutorials and tips  as a self-paced course to learn the software thoroughly or can jump into any section:

<https://www.qgistutorials.com/en/>

It is recommended to check the topics:

* Making a Map,
* Working with Attributes,
* Importing Spreadsheets or CSV files,
* Using Plugins,
* Searching and Downloading OpenStreetMap Data)

## QGIS video tutorials

* QGIS Interface: <http://qgis-tutorials.mangomap.com/post/79334660226/qgis-video-tutorials-module-1-the-interface#disqus_thread>
* Creating a Basic Map in QGIS:<http://qgis-tutorials.mangomap.com/post/82295156067/qgis-video-tutorials-module-2-creating-a-basic>
* Creating Vector Data: <https://vimeo.com/94646205>

## Daily Task 03/29/2022

In today's daily task we will do another project that will lead to the creation of a map through GIS but by combining many sources and extending our knowledge. Our goal is to identify which states have had more births than deaths in 2018 and create a map. We will also insert some landmark locations.

**a)** Please download everything in the folder and make sure you keep the same structure:

<https://drive.google.com/drive/folders/16GngfOxot26z3WnFJqhC_2uyCldr5syl?usp=sharing>

**b)** We will also be going to use data from here: <https://public-nps.opendata.arcgis.com/>

This site contains shape files for various attributes of National Parks. The one I am using is already in the folder above, but we will see how to download it if needed. Navigate to the Boundaries category and find the nps boundary dataset. In the screen that follows download the shape file.

**c)**  Follow along with the presentation of creating maps for various types of National Sites.

**d)** Now let's switch gears and present some information about Deaths and Births for the various states. The data comes from the National Vital Statistics System maintained by CDC here:

<https://www.cdc.gov/nchs/nvss/vsrr/provisional-tables.htm>

No need to download anything here we will do it with R!

**e)** Let's go over the TG\_Data\_1\_1.R. Make sure you rename it using your initials and put comments whenever you can.

**f)** Let's create a map with the various death vs births both with categorized and graduated tools.

**g)** After all this we will see an excellent tutorial by a former student Hugh Bullard on data coming from agriculture here:

<https://drive.google.com/drive/folders/1dPadT6oRXJ0YiQAjYs31117W5CIeRi6Y?usp=sharing>

## Shapefile Databases

* US Census Bureau: <https://www.census.gov/geographies.html>
* IRMA Data Store: <https://irma.nps.gov/DataStore>
* National Park Service: <https://public-nps.opendata.arcgis.com/>
* TIGER/Line Shapefiles: <https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.html>

# ArcGIS

In this Section we will explore [ArcGIS](https://www.arcgis.com/index.html) the go to software for map creation and geolocated data exploration. Unfortunately, ArcGIS is quite expensive so our only interaction will happen through the VLAB's resources.

## ArcGIS Starting Kit

Unfortunately, ArcGis is not a freeware like all the rest of the things we have seen so far.

BUT! We can always use the VLab to get to it. There is a whole desktop for ArcGIS found here:

<https://grok.lsu.edu/Article.aspx?articleId=16074>

Additionally, I was informed that the Ag Center has some licenses for ArcGIS, and it can be installed into your computers, but you have to request it

and the request has to be approved.

Coordinate with your adviser and the LSU Agcenter helpdesk if you want that option.

## Introduction to ArcGIS

Thanks to Ms Niu Huizhen, a researcher at the AG Econ Department we have some excellent resources for ArcGIS. Thank you Niu

* <https://www.youtube.com/watch?v=p4NbRw3QkGk&t=2s>
* <https://www.youtube.com/watch?v=hqHCJUudPvs&list=PL63EB94891DE02AA9>
* <https://www.youtube.com/watch?v=DkGmRv5cHlQ&t=13s>

## Basic commands in ArcGis

* <https://www.youtube.com/watch?v=KIy4xw8fit0&t=2s>
* <https://learn.arcgis.com/en/projects/get-started-with-arcmap/>

## Daily Task 2/19/2019

Today's task will help us dive into the ins and outs of ArcGIS. As mentioned earlier, this is not a free software, but instead we will use the vlab to complete the task, so we need to remember how that works.

**a)** Connect to the VLab and choose the ArcGIS desktop. If you need a refresher on how to do that check the section, **V-lab, Beyond Excell.**

**b)** While connected, navigate back to moodle and this page.

**c)** Connect to the census.gov site and go to this page: <https://www.census.gov/geo/maps-data/data/tiger.html>

d) Find and click the link: TIGER/Line Shapefiles - New 2017 Shapefiles here <https://www.census.gov/geo/maps-data/data/tiger-line.html>

**e)** On the tab Download, click on the option Web Interface

**f)** Select Year 2017, and layer type States (and equivalent) and click submit

**g)** Download the national file and save it in a repository on the folder ArcGIS in your documents. Call the file, ArcGIS Example and use our established naming system. The file is probably compressed, so make sure you expand it in the created folder. You should be looking at 8 files, one of which is a "shape file" .SHP. For a quick overview of what a shape file is please check here:

<https://en.wikipedia.org/wiki/Shapefile>

**h)** In the same folder there is a dbf file. We will check that later using R.

**i)** Start the program ArcMap (search through windows button bottom left). Click file and create New Document. Choose the template Traditional Layout and then the template USA and click ok.

**j)** Find the button "Add data" and navigate to the folder you created. Select the shp file and click add.

**k)** Let’s go back to our desktops and select the dbf file. Open it up with R using the code found here:

<https://drive.google.com/drive/folders/1V4w6wAkFzWJy_eJBez-g2AR9-XIEu606?usp=sharing>

**l)** Create the extra columns Char\_land and Char Water and over-write the dbf file.

**j)** Use the new dbf file and select the states with above average land area and above average water area in ArcGIS.

## Daily Task 2/25/2019

Today we will expand on ArcGIS and see how we can start visualizing a few more things using it.

But before we do that, lets extend the attributes we have for each state.

**a)** Follow this link to get to the repository connected to today's task:

<https://drive.google.com/drive/folders/1V4w6wAkFzWJy_eJBez-g2AR9-XIEu606?usp=sharing>

**b)** Download everything into a folder on your pc. We will be merging last week's dataset with yesterday's dataset.

**c)** Run the R script Analysis\_1\_3. Make sure a summary txt and a dbf file are produced.

**d)** Copy the folder tl\_2017\_us\_state into a drive on your vlab. Copy the file tl\_2017\_us\_state.dbf in that folder and override the old one. Make sure you also copy the summary txt on the vlab

**e)** Let's open up ArcGIS and load again a USA template. Add the shp file found in the folder you uploaded as a layer.

**f)** Let's explore now the properties of the added layer. Click that and go to the tab Symbology. Explore by Categories Quantities and Charts

**g)** Lets now open the Attribute table and do various selections.

**h)** Create a new column (Add Field) called Emp\_16\_Per. Select that new column and right click until you find Field calculator. We want this column to be the percentage of employed people in 2016. So basically, what we want is 100-UR\_16.

**i)** Let's create a pie chart of the employment vs unemployment in 2016

**j)** Let's create a bar chart of bachelor's vs high school in 2016

**k)** Let's now make various plots using quantities for poverty, unemployment 2016, unemployment 2017.

**l)** Let's now export the maps as pdf's

**m)** Let's now use the summary.txt to select and visualize various states based on poverty, unemployment, and education.

# IMAGE PROCESSING

Image processing is an integral part of data analysis that sometimes is overlooked since traditional statistics doesn't always have the means to work with images.

In Digital Ag image processing is needed when we are reviewing satellite images, drone images, but also images collected at the lab and in the field with conventional cameras.

The images could be focusing on different spectra and need not be always visually interpret-able.  In this section we will discuss general image processing ideas and relevant tools.

## Imager- an R package for image processing

Imager is a library in R useful for various image processing tasks. Imager is built on the [Cimg Library](http://cimg.eu/)

It can be used to perform many basic functions on an image, including statistical analysis. One of it's drawbacks is that it is memory inefficient, but it makes up for that with its versatility and ease of use.

* Basics on Imager: <https://cran.r-project.org/web/packages/imager/vignettes/gettingstarted.html>
* Different functionalities of [Imager: https://dahtah.github.io/imager/](https://dahtah.github.io/imager/)
* Rdrr/io's take on imager: <https://rdrr.io/cran/imager/>

## Daily Task 04/04/2022

* In this daily task we will try to do some image processing using our favorite R!
* Download the files in the following folder:

<https://drive.google.com/drive/folders/1p_usOMhy_9XJq5cmeAWoopGfDr_ZaLFN?usp=sharing>

* We will be working with the code TG\_ImageBasics\_1\_1. Make sure you create your own version.
* Installing imager (the basic package) might be a bit tricky!
* Please follow along and try to put as many comments as you can in the codes!

# IMAGES IN PYTHON

In this section we will continue our intro to image analysis and see how that can be achieved through Python.   
Again, the idea is to figure out what is possible (and maybe what is not) and practice a bit more our Python skills especially when it comes to image processing.

## Basic Libraries and their links

There are many libraries used in image analysis for python, but the three most prominent ones are:

* [OpenCV](https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html),
* [Scikit-image](https://scikit-image.org/)
* [Pillow](https://pillow.readthedocs.io/en/5.3.x/)

Depending on the application one might be better than the other. Ultimately it is a preference in coding style. We will, as usual, be agnostic about it and use whatever is more convenient.

We will also be using 3 image filters that are very common in the field for object detection (and more):

* [The Laplacian of Gaussian](https://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm)
* [The Difference of Gaussian](http://fourier.eng.hmc.edu/e161/lectures/gradient/node9.html)
* [The Determinant of Hessian](https://milania.de/blog/Introduction_to_the_Hessian_feature_detector_for_finding_blobs_in_an_image)

## Daily Task 04/11/2022

**A) Data and codes**

* Today we will be working with python and try to repeat the image processing we learned last week, but with a different library.
* The data and codes can be found here:

<https://drive.google.com/drive/folders/1dNCp5EE9ZMUuuIh1VyGsyEGYUeOh-4fo?usp=sharing>

* Please download everything and keep the structure
* We will need to install the library scikit image before we proceed, from the anaconda environment. Make sure you install it in the Digital Ag Environment.
* Make sure it installs all dependencies. This might take a while.
* We will also need to install the library cv2. This is the main image processing library in the core of the algorithm. Unfortunately, this might fail for the mac users...
* Make sure it installs all dependencies. This might take a while as well.

**B) Working with the code**

* We will first look at the TG\_Blob\_Count\_1\_1. This code requires jupyter notebook, so make sure you save it somewhere that is accessible by anaconda.
* This code will try to enhance the image detection we did with R and give us a count of the "blobs" present in the image.
* Make sure you add plenty of comments and ask questions!!

**C) Automating and extending**

* After we understand the idea behind blob count, we will apply it to multiple images at a time. Let's look into TG\_Blob\_Count\_2\_2. This code will be run on spyder so no need to worry about placement.
* This code will repeat the analysis we did in B but this time with multiple images and report a csv file of results.
* Again, make sure you add comments and ask questions.

# IMAGE STITCHING IN PYTHON

In this section we will introduce the idea of image stitching, a very important concept in digital agriculture, especially when using drones.

## Image stitching

Here you will find some basic links for image stitching:

[What is it?](https://www.ptgui.com/info/image_stitching.html)

[The wikipedia version](https://en.wikipedia.org/wiki/Image_stitching)

[Image panorama](https://towardsdatascience.com/image-panorama-stitching-with-opencv-2402bde6b46c)

## Daily Task 04/12/2022

In today's task we will be stitching various images to create a panorama. Please follow along and make sure you add as many explanations to the codes as you deem necessary.

A) Installing the packages

First we will install all the packages in python that we will need.

Select the Digital Ag directory and click on the play button selecting the "open terminal" option. This will take you to the directory that you have installed the Digital Ag directory through the anaconda prompt, making installing packages at the right spot a bit easier. What we need is the package imutils so type conda install -c conda-forge imutils and wait a bit. If that doesn't work, try pip install imutils.  If you already have the package close this and move to the next stop. If not, wait a bit until it loads.

After the imutils we need to install the package imageio. You probably have that already, but just in case you can install it in your environment the traditional way.

By the way, to see the packages installed already you can use the command conda list in the command prompt. Check to make sure that imutils is there and move on.

B) Working with the code

We will be following a version of the tutorial found here: <https://towardsdatascience.com/image-panorama-stitching-with-opencv-2402bde6b46c>

You can find the code and some images here:

<https://drive.google.com/open?id=1arpXk8b72oWFWMlomduMdZXNLoNTyTO0&authuser=gentimisth%40gmail.com&usp=drive_fs>

We will be following the TG\_ImageStiching\_2\_1.py. The TG\_ImageStiching\_1\_1 is the one found in the tutorial.

Add as many comments as you can and follow along!

# DATABASES, POSTGRESQL

## PostgreSQL-Installation and intro

Here you can find resources for postgreSQL our favorite database handler.

The following two links will help you install postgres and pgadmin, the essentials for running things ins postgresql

[Postgres windows installer](https://www.enterprisedb.com/downloads/postgres-postgresql-downloads)

[pgadmin the GUI to handle postgres](https://www.pgadmin.org/)

Some initial information about postgresql can be found below:

* [The official postgresql webpage](https://www.postgresql.org/)
* [What is postgreSQL](https://en.wikipedia.org/wiki/PostgreSQL)?
* [About postgreSQL](https://www.postgresql.org/about/)

## PostgreSQL tutorials

The following links can help you get on your way with postgresql

* [The official tutorial](http://www.postgresqltutorial.com/)
* [Introduction to SQL](https://www.w3schools.com/sql/sql_intro.asp)
* [Another intro to SQL](https://www.w3schools.com/sql/sql_intro.asp)
* [postgres cheatsheet](http://www.postgresqltutorial.com/wp-content/uploads/2018/03/PostgreSQL-Cheat-Sheet.pdf)
* [PosgreSQL 2021 (with mac install)](https://www.youtube.com/watch?v=DMl5fsc8PSk)

## Daily Task 04/18/2022

**A.** Installing Postrgresql

Let's first start with installing postgresql, the engine of database management. Please follow the link:

<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>

Select the version that best fits your machine and click download

When the program prompts you to install other components, select all of them

Provide the password postgres for both fields that need a password

Select port 5432

Select Default locale for Locale

Click next until the installation starts. This might take a while

**B.** Installing pgadmin4

Typically, Posgresql comes with pgadmin4, but just in case this did not work, you can download the latest version of pgadming 4 from here:

<https://www.pgadmin.org/download/>

Again, choose the appropriate version and hit install.

There shouldn't be any issues here, but just in case, check the installation video in the folder of this section if something goes wrong.

**C.** Downloading data and step by step instructions

Follow this link and let’s analyze together how we can read and merge multiple files with the same format using PostgreSQL:

<https://drive.google.com/drive/folders/1XFC0fINd6VWcE8Xk9eMEk1G8GwddYpL0?usp=sharing>

In the subfolder Postgress, there is a pdf document called Database\_Creation created by Namratha Mohan, a gifted former student of mine. The process should be the same, although Pgadmin has been updated (and re-written apparently) so perhaps there are some discrepancies.

Follow along and try to create a new database named: DigitalAg

Use the graphic user interface to include some of the data available in the folder in your database

Use the codes available in the postgress\_Codes word document to do basic database manipulation.

# DATA IN AGRICULTURE

## Available databases

In this page I have collected free open-source databases which have an agriculture flavor.

With your help this part of the class will be populated with interesting databases for the future students.

* <https://www.data.gov/food/>
* <https://agricola.nal.usda.gov/>
* <http://agris.fao.org/agris-search/index.do>
* <http://www.fao.org/faostat/en/#home>

Here is an excellent tool to get information about various agricultural products

* <https://quickstats.nass.usda.gov/#D69E828B-75A1-3AE1-8198-56B6C7073183>

Here is an excellent repository for basic ArcGIS maps:

* <https://www.census.gov/geo/maps-data/data/tiger.html>

A great site to get information about seasonality of agricultural products

* <http://www.financial-spread-betting.com/Seasonality-commodities.html>

A database for free Satellite Images (thank you Fellipe):

* <https://earthexplorer.usgs.gov/>

## Daily Task 04/25/2022

Dear all,

This will be our first "exploration lab" where we will put to use all the things, we learned so far.

Bring in your laptops, your notes and lots of patience. We will be utilizing one of the online datasets available in this section and we will do all sorts of analyses to it. I am not sure how long this lab will take since I have never tried such a thing before, but we will figure it out together and learn as we go.

Here is a list of the things I would like you to do:

**a)** Get in groups of 3. I will help you organize. One person from the group will create a google folder and share it with the other two. All three of you should be working on the same folder.

**b)** Follow the link from Data.gov and navigate to this page:

<https://catalog.data.gov/dataset/county-level-data-sets>

Utilize the query tool and download the data Poverty, Population, Unemployment and Education (the csv versions)

**c)** For each dataset, keep only the columns corresponding to the years 2019, including the reference columns in the beginning. Name the files Poverty\_2019, Population\_2019, Unemployment\_2019 and Education\_2019

**d)** Create a SQL database and add the four tables that you created. Pay attention to the definition of the variables.

**e)** Create a SQL table by merging the four tables based on the FIPStxt column. Name that table Combined\_2019. Careful this process might need a bit of googling ...

**f)** Create a smaller table that contains only the States/counties whose unemployment level in 2019 is higher than the mean unemployment level of all States/counties in 2019. Call that table High\_Unemployment\_2019. You can use R for this.

**g)** Export that table into a csv file and make a scatterplot between poverty all and unemployment for the two years respectively using R on High\_Unemployment\_2019. Name the scatterplot appropriately and save it as a pdf

**h)** Do a summary statistic on all variables for each of the columns in the two subsets you have created.

**l)** Do a correlation matrix for all the columns in the subset that contains the States.

Congratulations, you are one step closer to becoming an expert in analyzing data.

# USEFUL LINKS-PRESENTATIONS AND VARIOUS OTHERS

In this area I will be adding information relevant to the course that doesn't precisely fall on any one category. I might ask you to add some content as well.

## Digital Ag Day Spring 2019

Dear all,  
Follow this link to see more about the conference of digital ag that happened Tuesday Jan 28th here:  
<https://www.lsuagcenter.com/profiles/rbogren/articles/page1549036492253>Videos of the talk can be found here:   
<http://tcs.lsuagcenter.net/tcs/#page:recordingList&pageNumber:1&id:DB1D1791-C696-4D3C-8A58-820A6B20AB11>Who knows maybe next time YOUR posters are up there!

## API's

In this page I will add various links to information about API's, one of the most useful developments in computer science with respect to human-machine and machine-machine interface.

* <https://www.youtube.com/watch?v=7YcW25PHnAA&t=1s>
* [The wiki page](https://en.wikipedia.org/wiki/Application_programming_interface)

1. **Class Recordings 2022**

01/18/2022, **Introduction**:

[**https://lsu.zoom.us/rec/share/dmarRMBpb71-\_p3o3s\_9uj\_32ksQW27VMc1VtWcz7kFHljndl5nSYBuRZr6gQS7A.TxsITeEgbKid2OAd**](https://lsu.zoom.us/rec/share/dmarRMBpb71-_p3o3s_9uj_32ksQW27VMc1VtWcz7kFHljndl5nSYBuRZr6gQS7A.TxsITeEgbKid2OAd)

01/25/2022, **R Intro:**

[**https://lsu.zoom.us/rec/share/zfFlP5syEtzqxZbJrlb25KlMrwxdgBJmUovuxKYSXlFxO0n4uduFwACdZphX6i6H.RVzSer6mRh-ywxKq?startTime=1643144686000**](https://lsu.zoom.us/rec/share/zfFlP5syEtzqxZbJrlb25KlMrwxdgBJmUovuxKYSXlFxO0n4uduFwACdZphX6i6H.RVzSer6mRh-ywxKq?startTime=1643144686000)

01/31/2022, **R Simple Data Manipulations:**

[**https://lsu.zoom.us/rec/share/Btom5m23b6c1oDgN-Mq94ffLI8UJntHi0Sg5e1\_1Er8D7kfH2I-8AGgoMS8ikPt4.d9QQoq3FkU6sI0h9?startTime=1643655573000**](https://lsu.zoom.us/rec/share/Btom5m23b6c1oDgN-Mq94ffLI8UJntHi0Sg5e1_1Er8D7kfH2I-8AGgoMS8ikPt4.d9QQoq3FkU6sI0h9?startTime=1643655573000)

[**https://lsu.zoom.us/rec/share/AAeg0IyFzfJUmSgO5uxki\_6riqOw3cDuE6sCQDxBeZ0daFpsUEoXb4\_eKVzwJb-z.gxAs23shCrH6gduL?startTime=1643660759000**](https://lsu.zoom.us/rec/share/AAeg0IyFzfJUmSgO5uxki_6riqOw3cDuE6sCQDxBeZ0daFpsUEoXb4_eKVzwJb-z.gxAs23shCrH6gduL?startTime=1643660759000)

02/01/2022, **Timeseries Intro:**

[**https://lsu.zoom.us/rec/share/SIYG\_O2zVzuIPjfBkQaLoaUXqiBzZde36KGXIFlM322y\_-dsWk6Ibh2PDFmt6tlS.-UkCp-mWgNcKjO3L?startTime=1643749672000**](https://lsu.zoom.us/rec/share/SIYG_O2zVzuIPjfBkQaLoaUXqiBzZde36KGXIFlM322y_-dsWk6Ibh2PDFmt6tlS.-UkCp-mWgNcKjO3L?startTime=1643749672000)

02/08/2022, **Timeseries Forecasting:**

[**https://lsu.zoom.us/rec/share/-mykjZCNRv\_gmjLrUh8Uxo5MnqzNSC7ccl0ZJc00A4SaMX1RL22yl1S3B0CJ-QtU.pvx9gq3KlV47Eq0e?startTime=1644260493000**](https://lsu.zoom.us/rec/share/-mykjZCNRv_gmjLrUh8Uxo5MnqzNSC7ccl0ZJc00A4SaMX1RL22yl1S3B0CJ-QtU.pvx9gq3KlV47Eq0e?startTime=1644260493000)

02/08/2022, **Timeseries and R Markdown:**

[**https://lsu.zoom.us/rec/share/QydW0jdLq8svPvIMNyV-3JCBU-b0IVFRc1hdg6SluuDPzOCXm4soy1MdySbQxlKw.vNTSHppP1xI2L1ay?startTime=1644354305000**](https://lsu.zoom.us/rec/share/QydW0jdLq8svPvIMNyV-3JCBU-b0IVFRc1hdg6SluuDPzOCXm4soy1MdySbQxlKw.vNTSHppP1xI2L1ay?startTime=1644354305000)

02/14/2022, **Introduction to Python:**

[**https://lsu.zoom.us/rec/share/txY7vfENWrv0OYMcEXnVlZAs148\_1E3DjJJlqQaMxPWqBCVXGXvi8LrNsdJCUoDn.c6LsX\_Iouywc55Mg?startTime=1644865362000**](https://lsu.zoom.us/rec/share/txY7vfENWrv0OYMcEXnVlZAs148_1E3DjJJlqQaMxPWqBCVXGXvi8LrNsdJCUoDn.c6LsX_Iouywc55Mg?startTime=1644865362000)

02/15/2022, **Python and Coding:**

[**https://lsu.zoom.us/rec/share/MgT7bNA0\_j8fuE5hs5gx-5CIo7\_XjNErJPbiGW00Ctj6FLkdkpJjFBbDjF5I-nw3.RPP10HNYsKG1bfwg?startTime=1644959199000**](https://lsu.zoom.us/rec/share/MgT7bNA0_j8fuE5hs5gx-5CIo7_XjNErJPbiGW00Ctj6FLkdkpJjFBbDjF5I-nw3.RPP10HNYsKG1bfwg?startTime=1644959199000)

02/22/2022, **Neural Networks Specifics:**

[**https://lsu.zoom.us/rec/share/TtPfAxx83EOY6TEqdhyeQTx1nkcu5N\_oEAlXF3o6DK\_6tRHt3ECM\_YfzpCC-uYWo.MWozEsHCdnMd9D7p?startTime=1645563869000**](https://lsu.zoom.us/rec/share/TtPfAxx83EOY6TEqdhyeQTx1nkcu5N_oEAlXF3o6DK_6tRHt3ECM_YfzpCC-uYWo.MWozEsHCdnMd9D7p?startTime=1645563869000)

03/07/2022,**Support Vector Machines:**

[**https://lsu.zoom.us/rec/share/h6ua6ahqn4T9jYWu2RqlICGc\_dHyYPx\_CbSNAEBMFa280koVLmz-sh3uULTJFKyr.X1miqlz-kvO8ERyf?startTime=1646680030000**](https://lsu.zoom.us/rec/share/h6ua6ahqn4T9jYWu2RqlICGc_dHyYPx_CbSNAEBMFa280koVLmz-sh3uULTJFKyr.X1miqlz-kvO8ERyf?startTime=1646680030000)

03/08/2022, **Support Vector Machines:**

[**https://lsu.zoom.us/rec/share/OYrB67ECc68vzrZhzaMzfoc70a2zASGN1HX6qxBNlq4tq2WEVdM6MCKWQwjBSQHM.jvxWfsyWJPEazHZT**](https://nam02.safelinks.protection.outlook.com/?url=https%3A%2F%2Flsu.zoom.us%2Frec%2Fshare%2FOYrB67ECc68vzrZhzaMzfoc70a2zASGN1HX6qxBNlq4tq2WEVdM6MCKWQwjBSQHM.jvxWfsyWJPEazHZT&data=04%7C01%7Ctgentimis%40agcenter.lsu.edu%7C120e57fdf58b4fbe97a708da0151df84%7C804b509899084bdf9c06b3df777563aa%7C0%7C0%7C637823748418134049%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000&sdata=wy2wwqIJv%2FM9GseDHeIZNVDsYC9L4wU0cSYCf%2BHnyho%3D&reserved=0)

03/22/2022, **Random Forests 1:**

[**https://lsu.zoom.us/rec/share/FIabjscP4WtwHQcwFEvN4-R7kY4zrAhWOz86XUWXCDlPVIf2dy2BARX6xFZKM\_Z0.KesoTtBUus7vCSjf?startTime=1647979818000**](https://lsu.zoom.us/rec/share/FIabjscP4WtwHQcwFEvN4-R7kY4zrAhWOz86XUWXCDlPVIf2dy2BARX6xFZKM_Z0.KesoTtBUus7vCSjf?startTime=1647979818000)

03/25/2022, **Random Forests 2:**

[**https://lsu.zoom.us/rec/share/JfcoKbOaOfTtyH25MywmJ4BXahTzyY7CvPEnto5rHt0veWSOmYFAgGBmT4z5J3Kv.n4\_FhWhCChDRMT4b**](https://lsu.zoom.us/rec/share/JfcoKbOaOfTtyH25MywmJ4BXahTzyY7CvPEnto5rHt0veWSOmYFAgGBmT4z5J3Kv.n4_FhWhCChDRMT4b)

03/28/2022, **Machine Learning Wrap Up:**

[**https://lsu.zoom.us/rec/share/6r1ozPNxhxGXJOZdN17n2yP-oxA9yRizRXH0-lgUHttvDdq-rM-fcGhx8wnUuIfo.f0jyFksw0-BOVtIC?startTime=1648490468000**](https://lsu.zoom.us/rec/share/6r1ozPNxhxGXJOZdN17n2yP-oxA9yRizRXH0-lgUHttvDdq-rM-fcGhx8wnUuIfo.f0jyFksw0-BOVtIC?startTime=1648490468000)

03/29/2022, **QGIS introduction:**

[**https://lsu.zoom.us/rec/share/FA\_\_nL8oQTdKI\_05529tTt9RY7ztQ\_IAf2HErcSE6fbyPdA47rSfMy3Ogr-C6VFa.U5iSs1vpdyZNgOmI?startTime=1648584208000**](https://lsu.zoom.us/rec/share/FA__nL8oQTdKI_05529tTt9RY7ztQ_IAf2HErcSE6fbyPdA47rSfMy3Ogr-C6VFa.U5iSs1vpdyZNgOmI?startTime=1648584208000)

04/04/2022, **Images in R (1):**

[**https://lsu.zoom.us/rec/share/QXP5JeGFD51YPuJAwnp3dsJ9eeyMijUwZkAHM5h47ZWPr9GqAH8kqZB\_3oI0Zdkd.boAHVBfyd2wZ1FMJ?startTime=1649095549000**](https://lsu.zoom.us/rec/share/QXP5JeGFD51YPuJAwnp3dsJ9eeyMijUwZkAHM5h47ZWPr9GqAH8kqZB_3oI0Zdkd.boAHVBfyd2wZ1FMJ?startTime=1649095549000)

04/05/2022, **Images in R (2)**:

[**https://lsu.zoom.us/rec/share/u7hnUWcGms9uoZPCQIntC3pheUeabCLa-yrBakaxLtRwVnFsA4Ja0-2guoZCbdkd.W5yM-QljIhgEjHhn?startTime=1649189086000**](https://lsu.zoom.us/rec/share/u7hnUWcGms9uoZPCQIntC3pheUeabCLa-yrBakaxLtRwVnFsA4Ja0-2guoZCbdkd.W5yM-QljIhgEjHhn?startTime=1649189086000)

04/11/2022, **Images in Python:**

[**https://lsu.zoom.us/rec/share/D0QFnIpuL-31HPmTaAj\_DbcJyeDFG3MBKIpRgN4SaunldRs-mthhyzj8YOzYYj8E.ji8Q0QXFAhzBUYb4?startTime=1649700307000**](https://lsu.zoom.us/rec/share/D0QFnIpuL-31HPmTaAj_DbcJyeDFG3MBKIpRgN4SaunldRs-mthhyzj8YOzYYj8E.ji8Q0QXFAhzBUYb4?startTime=1649700307000)

04/12/2022, **Image Stitching:**

[**https://lsu.zoom.us/rec/share/TFFvwn3WfM0je1XRLBfwgKoY0BdX1kOkdcsZ0f1dkNESIXlQa4RW5UIJgXhcjFbh.kBiVX6FFtXTm6D6X?startTime=1649794172000**](https://lsu.zoom.us/rec/share/TFFvwn3WfM0je1XRLBfwgKoY0BdX1kOkdcsZ0f1dkNESIXlQa4RW5UIJgXhcjFbh.kBiVX6FFtXTm6D6X?startTime=1649794172000)

04/25/2022, **PostgresSQL Example:**

[**https://lsu.zoom.us/rec/share/uwOqwMIaVjiyYeXN6FqVU-ubapf-VQdxYDVZmBsPxlqm4a1fio90D3ipoSmaddPC.VQGx1HN\_IU1P9Ejn?startTime=1650909929000**](https://lsu.zoom.us/rec/share/uwOqwMIaVjiyYeXN6FqVU-ubapf-VQdxYDVZmBsPxlqm4a1fio90D3ipoSmaddPC.VQGx1HN_IU1P9Ejn?startTime=1650909929000)

04/26/2022, **Convolutional Neural Networks:**

[**https://lsu.zoom.us/rec/share/A-JU7gmfhLhBLp2Cskpr99A-1YqUfI8AHn2fRUOjWfHbeOPrHetNVs\_slDq\_zAC5.j9mVhrisZ811rann?startTime=1651003489000**](https://lsu.zoom.us/rec/share/A-JU7gmfhLhBLp2Cskpr99A-1YqUfI8AHn2fRUOjWfHbeOPrHetNVs_slDq_zAC5.j9mVhrisZ811rann?startTime=1651003489000)

05/09/2022,**Presentations:**

[**https://lsu.zoom.us/rec/share/EivJqWkRnbsKLyXAsVs9AVaoBxsv478neT8rRz04UzTLvpc29YkTu1QUV0khe\_kW.tSF9zOAE9ktP1xop?startTime=1652120277000**](https://lsu.zoom.us/rec/share/EivJqWkRnbsKLyXAsVs9AVaoBxsv478neT8rRz04UzTLvpc29YkTu1QUV0khe_kW.tSF9zOAE9ktP1xop?startTime=1652120277000)