Week_1

04 August 2022 14:18

Week_1

1) Discover of Dataset - Public (free), Private (paid), Personal(Your Device data)

Public : Awesome dataset —> not for search
Google dataset search
Kaggle dataset
Data.gov. → Owned and published data by govt
Datameet __ community

Private dataset: Corporate → most invested dataset
Paid dataset

Personal dataset → Mobile app dataset

Types

Structured dataset → you know schema, no ambiguity
Table from a sheet/database
Database

Semi Structured dataset → Json file

Unstructured dataset → you know nothing about schema
Pics, piece of text.

- Not binary, but continuous.
- Possible to take unstructured convert to structured

Structured → dataset
Spreadsheet Couples str
Shape file (GAMP – geographical data) → Spatial into as used as tabular info in a single consume.

Semi Structured → Document - pdf
Wikipedia page

Values:

Categorical: → allows fewer operations

List, sort not many operations
For ex colors like red, blue

→ you can infer information. Ex. boolean (True, False)

→ unordered (red , blue).. Cant define big or small

→ names of places

→ ordered (low, medium, high)

→ unstructured (text, binary)

Numerical: → series of operation

→ add, subtract, take ratios, perform several desired operations.

→ number values like int, float

→ ex int (-2,-1,0,1,2)

→ real num, natural number, whole number etc

Composite: → even more operations as composite comprise of __ multiple element

→ ex. It can contain an array which have list of numerical values and categorical values.

→ ex. date/time, spatial structured (lat,log,shape), specialised (ip,currency)

1.4 and 1.5

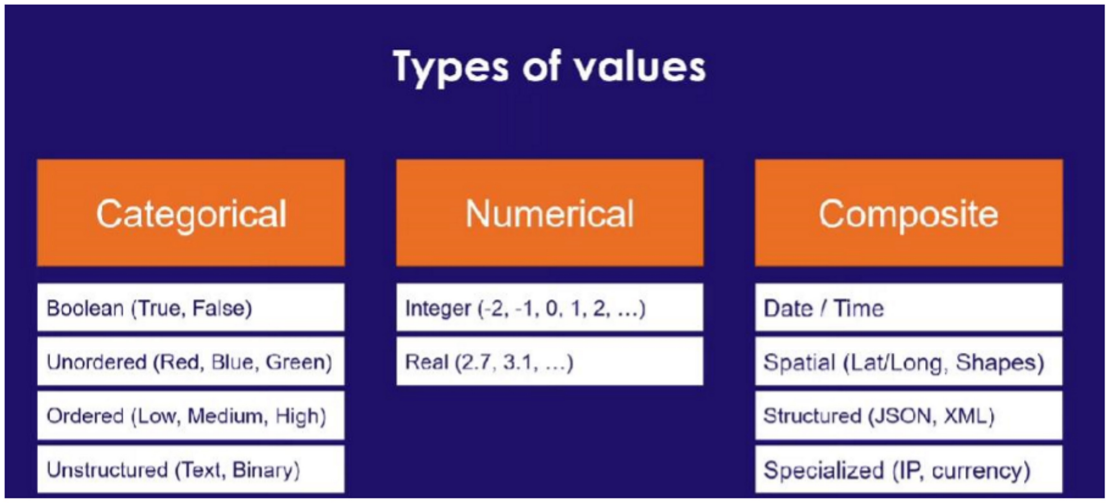
04 August 2022 14:14

1.4

Types of values

3 types-

1) Categorical- can only list and sort.
   Eg Colors
2) Numerical- Real numbers or integers. Add,multiply,take ratios etc.
3) Composite- Comrises of multiple elements.
   Eg. An array of numerical or mixed(both cat and num) values



In unstructured – text,binary there can be anything like an image or video while in structured there are files which have internal structure (json or xml document)

video while in structured there are files which have internal structure (json or xml document).

Spatial (Shapes)- This means a set of points – eg ABCD (A rectangle made of 4 points)

1.5

Understanding data tells us which dataset is easier to work with-

Structured better than unstructured- We do not need to extract information.

Numerical better than Composite and categorical.

2.1

Filename.tsv- tab separated value

F12 button is used to open network inspector.

3 ways to get data-

1) **Download- eg Kaggle/imdb**
2) **Query the data-**It may be on a database, it may be available through an API or a library, but these are ways in which you can selectively query parts of the data and stitch it together.
   Google has an Undocumented API
3) **Scrape data- When both querying and downloading fail.**
   Data is not directly available in a convenient form that you can query or download, but it is in fact on a web page, PDF file or an excel file.

We have used Beautiful Soup(a python library) in this Module.

week3

04 August 2022 14:28

Topic	Key Points
Data cleaning in excel	❖ After getting the data,it needs to be converted into an effective form so that it can be used for analysis. Here,data cleaning comes in picture.Following are some key points to keep in mind while cleaning the data-- 1.Find and replace 2.Changing the data format 3. Remove extra spaces • This is done using “Trim Function” • Trim function removes extra spaces from the cell • Format: =Trim (Cell Value / Text) 4. Selecting blank cells and deleting their entire row • Done using “Find and Select” 5.Remove duplicates • Done using function “Remove duplicates” in Data tab
Data Transformation in excel	1.Calculating ratios • Format=(col1/col2) • For rows having blank values in col2, we get ‘ #VALUE!’ 2.Pivot table • tool to calculate, summarize, and analyse data

	2.Pivot table • tool to calculate, summarize, and analyse data and lets you see comparisons, patterns, and trends in your data.
Converting text-to-columns in excel	• Done using ”text to column” function in data tab • Splits the data into multiple columns based on delimiters (semicolon, comma, space etc) or based on fixed length
Data aggregation	• Data aggregation is done to provide data summaries which help in examining trends, making comparisions and to reveal insights of data.

	<u>EXCEL FEATURES TO VISUALIZE AGGREGATED DATA:</u> **1.** <u>Colour Scales</u> ▪ It helps to identify clusters in data. **2.** <u>Pivot Table</u> **3.** <u>Spark lines</u> ▪ Spark lines are used to provide the visual representation to show the trends in data and to highlight the minimum and maximum values. **4.** <u>Data bars</u> ▪ Data bars provide a very easy and quick way for graphical illustrations of numerical columns.

TDS Week 4a

04 August 2022 14:19

4.a (Model the Data)

4.1 Introduction (04:18min)

1. Excel, Python & Others

Excel: Correlation, Regression & Outlier detection.

Correlation: Correlation is a statistical measure that expresses the extent to which two variables are linearly related

Regression: Regression is a statistical method that attempts to determine the strength and character of the relationship between one dependent variable and a series of other independent variables.

Python: Classification, Forecasting, Clustering

Others: R/RStudio, Rattle, PyCaret

Precheck: Excel File--> Options --> Add-ins --> Manage (Excel Add-ins) --> Select Analysis ToolPak

4.2 Model the data: Correlation with Excel

Process: Data --> Data Analysis --> Correlation --> Select Input range (variable values to be considered for correlation) --> give any cell as output range then OK

We will get a square matrix of variables having values in a lower triangle with diagonal values 1 (represent self-correlation). Generally, >0.75 (more than 75%) shows a higher correlation between variables.

We will get a square matrix of variables having values in a lower triangle with diagonal values 1 (represent self-correlation). Generally, >0.75 (more than 75%) shows a higher correlation between variables.

Correlation can be positive & negative. If variables move in the same direction, it will have a positive correlation, and the number shows the magnitude. e.g., If fast food consumption and health issues are positively correlated variables.

Similarly, sales volume and sales price is negatively correlated variables. means an increase in the price of an item leads to decreased sales.

p.s. 0 or close to 0 represents no or less correlation between variables.

Then we can insert graphs (based on data from column) & trend lines to visualize.

4.3 Model the data: Regression with Excel

Process: Data --> Data Analysis --> Regression --> Input Y range (Dependant Variable) --> Input X Range (Independent variables) --> Output New sheet

Result Interpretation:

Adjusted R-Square: It represents the % of variations explained in dependent variables by independent variables.

Significance values: <0.05 then good model

P-values: These represent the significance value of an individual independent variable. <0.05 then independent variables should be included in the model.

Math model: Dependant variable = Coeff (1) * independent var (1) + ... + Coeff(n)* independent var(n)

Coefficient: It represents the impact of an independent variable (keeping all other independent variables constant) on the dependent variable.

4.4 Outlier detection with Excel:

4.4 Outlier detection with Excel:

Q1: First quartile

Q3: Third quartile

IQR: Inter quartile range (Q3-Q1)

Lower Bound: Q1-(1.5*IQR)

Upper Bound: Q3+(1.5*IQR)

Excel Function: Quartile.exec(array, quartile number (1/2/3/4))

Outlier: (<Lower bound) or (>Upper bound)

Visualization: One can plot data points using Box-Whisker.

4.5 4.6

04 August 2022 14:20

4.5
Forecasting using python

Important libraries used

```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
from sklearn.metrics import mean_absolute_error
from pandas.plotting import autocorrelation_plot
from statsmodels.tsa.arima_model import ARIMA
```

ARIMA module is popular for forecasting data

In lecture rolling mean technique was used which means
***Rolling_mean5 means collecting the mean of past 5 days ***
Eg if today is friday then i will collect the data from mon,tue,wed,thur,frid
Next on sat i will take avgs of tue,wed,thur,fri,sat

To know the best value for rolling_mean5/10/15 we use autocorrelation model

We find that 2/3/4or5 days are good others are not so good

We use mean squared error to compare how good the predicted values were and found out that 5 day had the least sum squared error as we increase the days sse also increases which is bad

ARIMA also suggests that 5 day is the best

ARIMA stands for autoregressive integrated moving average

ARIMA has 3 parameters arima(p,d,q)
P:number of lag information
d:degree of differencing
q:size/order of moving averages

4.6

L4.6. Data classification with Python

Problem statement;- based on previous data(application data we need to classify whether to approve a credit card application for a particular customer or not.using decision tree

Step 1 :-
Step 2 : - we are now making only 2 variables 0,1

0 for all those who have one month due,or no due at all
1 for all thoes who are having due for more than more than 1 month

Step 3 :- converting categorical var to numeric variable

we use ordinal encoder for ordinal categorical variable(having a order)

We use label encoder for categorical variable without order

We use pandas dummy for binary features (male=0 ,female=1

Removal of outliers

For amount income total we used the IQR technique

Create the 75th and 25th quartiles
IQR = 75th quartile - 25th quartile
Lower range = -1.5*IQR
Upper range = +1.5*IQR
Every amount income total should fall in between the lower and upper range
All the points which don't fall in this range are removed from the data set

For count children and count family members visually we can see and set that any value >8
Is an outlier and is removed from the dataset

Modeling phase

We use to slip our data set into test and train sets it takes 1 parameter the size , train_test_split(x,y,test_size=0.2)

We use min_max_scalar as transformer

SMOTE (GOOGLE THIS) based in k nearest neighbour
BASICALLY WHAT IT DOES IS BALANCE THE DATA SET BY SYNTHETICALLY CREATING DATA POINTS FOR THE MINORITY CLASS TO REDUCE THE IMBALANCE IN THE DATA SET

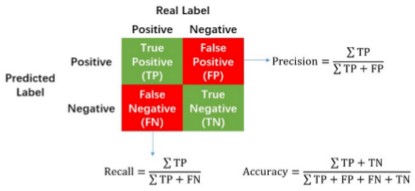
0----> 27356 AFTER SMOTE_____> 0—------->27356
1---->581 1—------->27356

Feature_importance_,gini_importance is on which decision tree split occurs

Pydotplus gives the entire decision tree

Module 5a

04 August 2022 14:22

Module	Title	Notes	Additional Resources
Module 5a			
5.1	Sentiment analysis with Excel and Azure ML	- Azure Machine Learning Add-in present in Excel to do sentiment analysis - IMDB movie review dataset opened in excel (Review of the movie (as tweet_text) , Positive/Negative label as sentiment) - Text Sentiment Analysis - - Input : Tweet Text - - **Output** : **Sentiment, Score** - - Procedure : Select all the tweets (movie reviews) as input >> Select the output Column >> Click Predict - - The Sentiment can be either positive or negative - - The scores associated with the setiment is given as a number. It can be converted to a percentage. - - - If score is closer to 0 - Sentiment is negative - - - If score is closer to 100 - Sentiment is positive - - Pivot table to create a **confusion matrix** to analyse classification performance - - - Actual labels as Row - - - Predicted Labels (Sentiment) as Column - - Metrics such as preciscion, recall and accuracy can be computed - - - 4 numbers in the 2X2 confusion matrix are True and False Positive (TP and FP), True and False Negative (TN and FN) **- - - Accuracy = TP+TN / TP+FP+TN+FN** - - - - Sum of diognal elements / Sum of all the elements in the confusion matrix **- - - Preciscion = TP / TP+FP** - - - - Of all the positives our algorithm predicted x % as positive **- - - Recall = TP / TP+FN** - - - - Of all the negatives our algorithm predicted x % rightly as negative - - Can use the above metrics to compare Azure ML and Python Text Blob - - Excel can give memory error when processing too many rows. This problem is not there while using python.	

5.2	Sentiment analysis with Python and SpaCy	- Sentiment prediction for text using python
		- Open source sentiment toolkit - TextBlob (derived from google)
		- review and sentiment columns are available.
		- - Input : Movie Review
		- - Output : **Subjectivity (range : 0 to 1), Polarity (range: -1 to +1)**
		- - Eg : Subjectivity score may be high for "You are very beautiful" because "beauty" is subjective
		- - Code sample
		`data[`TextBlob_Subjectivity`] = data[`review`].apply(lambda x : TextBlob(x).sentiment.subjectivity)`
		`data[`TextBlob_Analysis`] = data[`TextBlob_Polarity`].apply(lambda x : "negative" if x<0 else "positive")`
		- - Generate **Confusion matrix** using
		`classification_report(data[`sentiment`],data[`TextBlob_Analysis`])`
		- - f1-score is a combination of preciscion and recall	

5.3	Geospatial analysis with Excel	Video by Anand S "Manhattan's Coffee Kings" - A Starbucks/McDonalds case study - - How far one has to walk for nearest coffee in Manhattan - - 230 Starbucks and 57 McDonalds in Manhattan - - When more customers are close to a coffee shop it makes the shops more profitable - - Moving shops to make them more profitable.	Reference Links Below video: - https://blog.gramener.com/the-making-of-manhattans-coffee-kings/ - https://blog.gramener.com/shaping-and-merging-maps/ - https://blog.gramener.com/visualizing-data-on-3d-maps/ - https://blog.gramener.com/physical-and-digital-3d-maps/

5.4	Geospatial analysis with Python (GeoPandas)	Geospatial analysis using Starbucks and McDonalds store coordinates in New York **import folium** import **geopy.distance** - - Get the data into a Pandas dataframe using read_csv - - Merge the latitude and longitude columns as a tuple into one columns called Coordinates - - Obtain latitude and longitude of Empire State Building and store it in NY_coord - - Compute distance of the stores using geopy.distance.distance(NY_coord,row.Coordinate).km - This gives the Haversine distance - the angular distance between two points on the surface of a sphere - - Visualize the data on a map using the **folium library** m = folium.Map(location=[lat,lon],zoom_start=10) - - input parameters : where to center the map and at what zoom level to start folium.Marker(location=[lat,lon],popup=store_name,icon=folium.Icon(color=color)).add_to(m) - - Can get stores at greater than/lesser than/ farthest/ closest distance df_farthest = df.iloc[df.groupby(`store`)[`Distance`].agg(pd.Series.idxmax)]	
Module 5b			
5.5	Geospatial analysis with QGIS		
5.6	Image classification with Python (Keras)		
5.7	Image auto classification with Google Cloud Vision		
5.8	Talk: Exploring the Movie Actor Network in Python		

week 6

04 August 2022 14:27

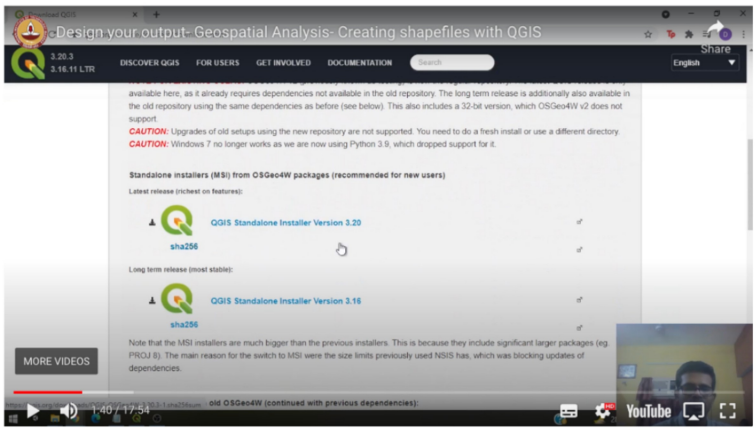
TDS Week 6

L1 Geospatial Analysis with QGIS

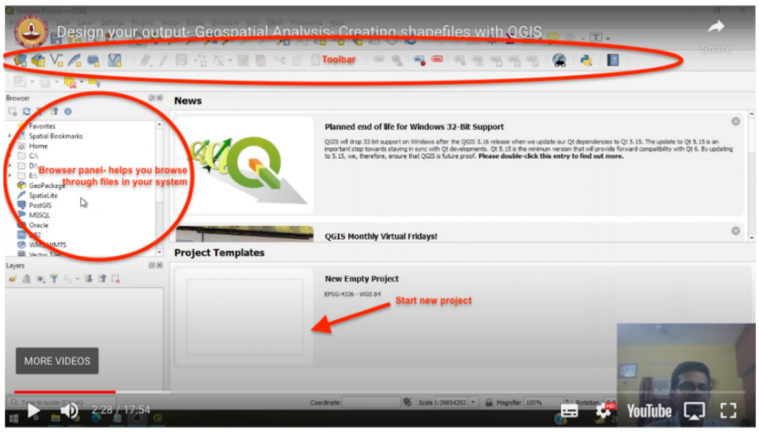
QGIS- Open source geographic information system to create shape files and KML files

Using QGIS

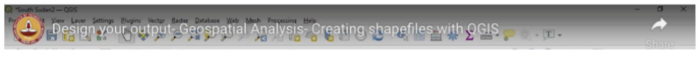
1. Download

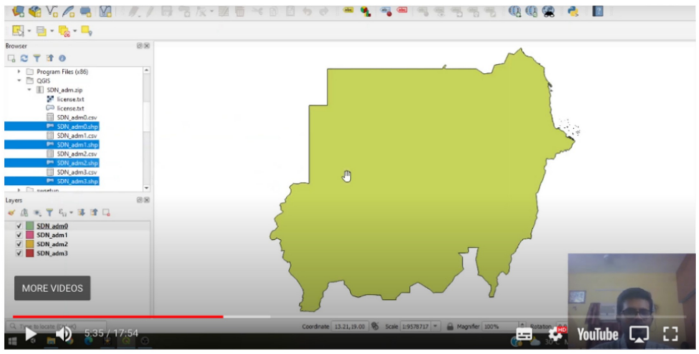


2. Intro Page



3. Getting Shape files:
   a. https://www.diva-gis.org
   b. go to “free spacial data” tab on above site → “country level data” → choose country of your choice → subject: “administrative areas” → click “ok” and download → zip file will be downloaded
   c. File is now downloaded and put in folder, it can be found in “browser panel” of QGIS software → when you open folder, csv and shape files are present → select all shape files only
   d. drag selected files from browser panel and drop it

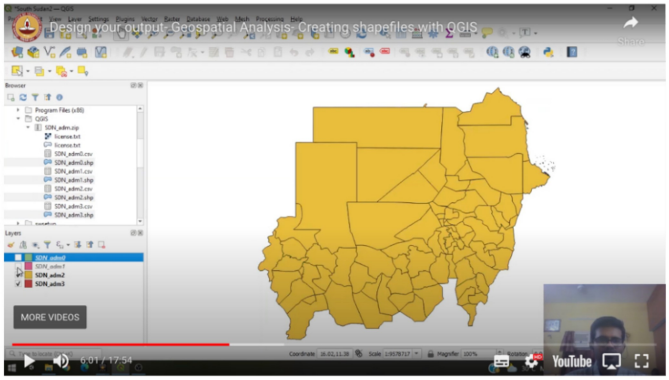




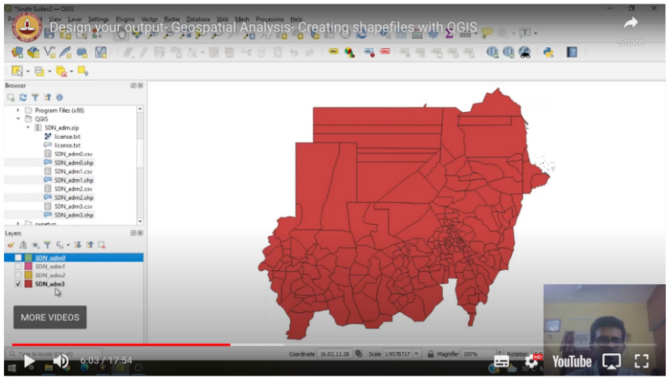
4. Playing with shape files:

   each shape file is layered one on top of the other, you can unselect the files.

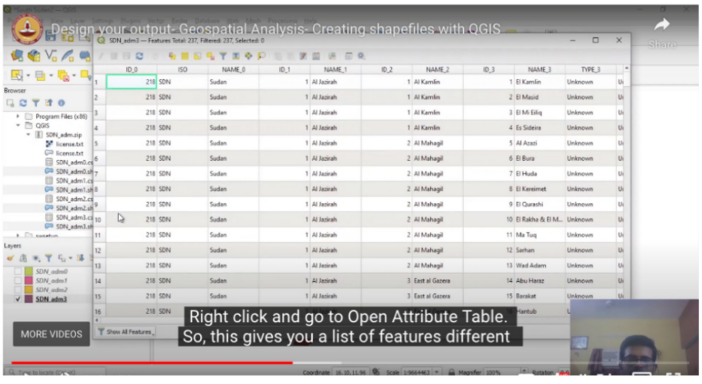
   - unselecting adm_1



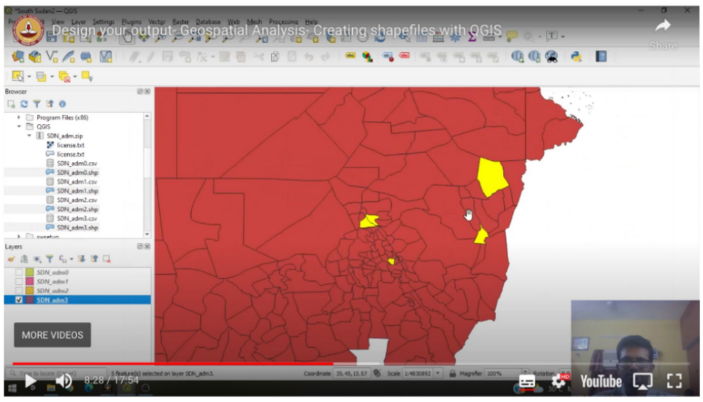
   - unselecting adm_2



   right click adm_3 → open attribute table

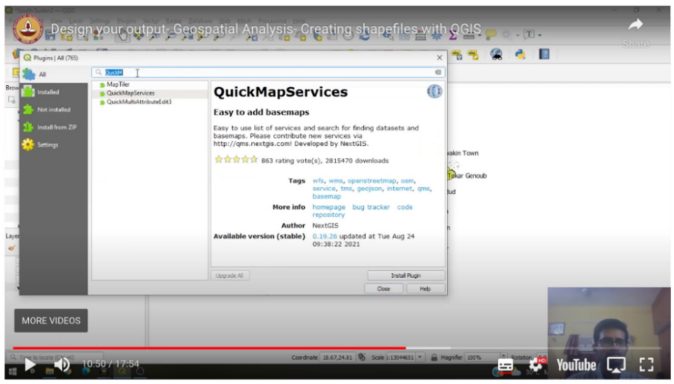


selecting rows in this, and minimising this table would highlight the region on the shape file like this:



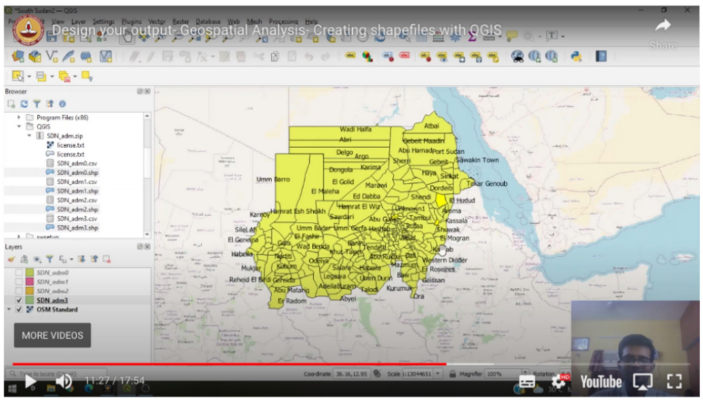
a. changing colour of shape file- right click adm_3 → properties → symbology → color
b. adding attributes to the map- right click adm_3 → properties → labels → single labels → select the label that you want to have on top of the map from the drop down menu
c. overlaying this shape file on top of the world map-

   go to plugins on top of your screen → manage and install plugins, you get this:

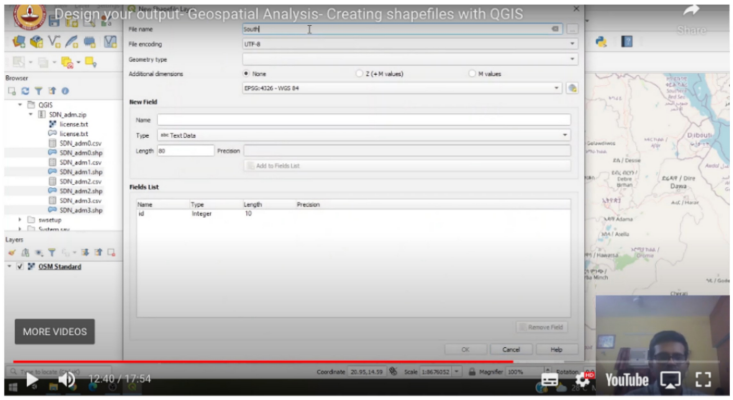


   → type "QuickMapServices" in the search bar→ select and install plugin

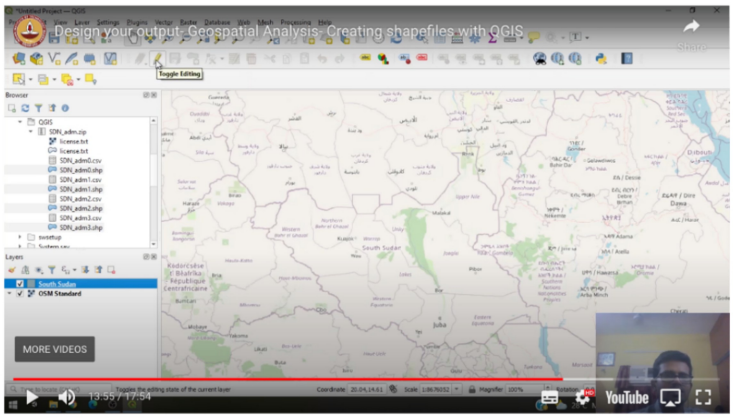
   go to web on top of your screen → QuickMapServices → OSM → OSM Standard, and you'll get this:



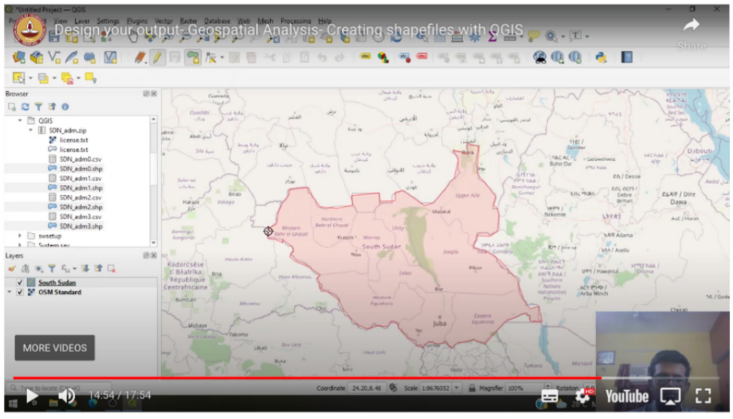
5. Creating a shape file:
   1. Remove all layers
   2. Go to "layer" on top left of your screen → create layer → new shape file layer



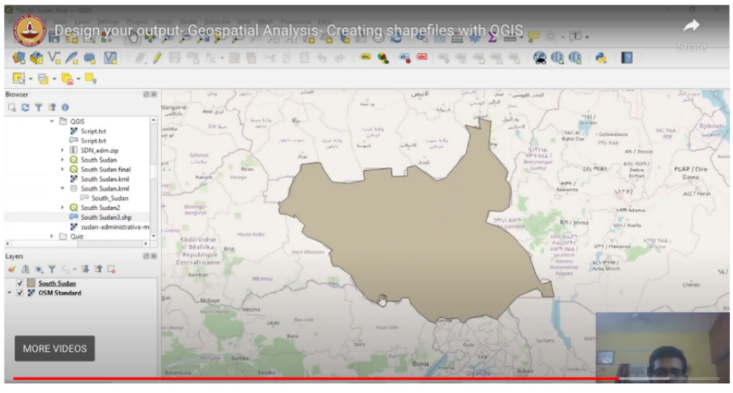
3. geometry type → polygon
4. new field → name → put in a name → “add to fields list” → ok
5. draw the border- go to toggle editing in toolbar



→ add polygon → left click on map and follow boundary of country→ result:

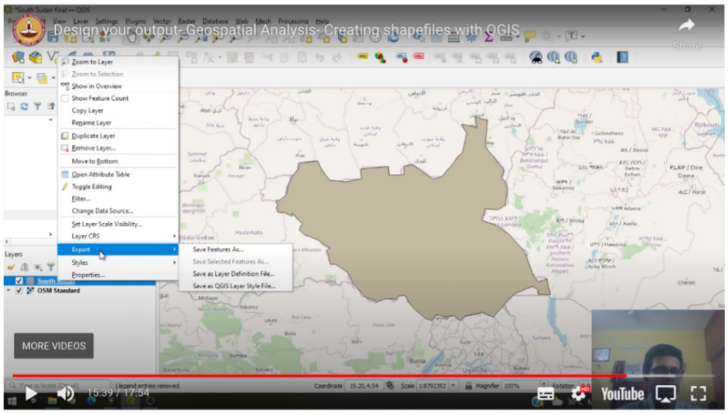


→ right click→ Put id = 1, Name = CountryName → ok



^ ^new country shape file

6. Exporting shape file

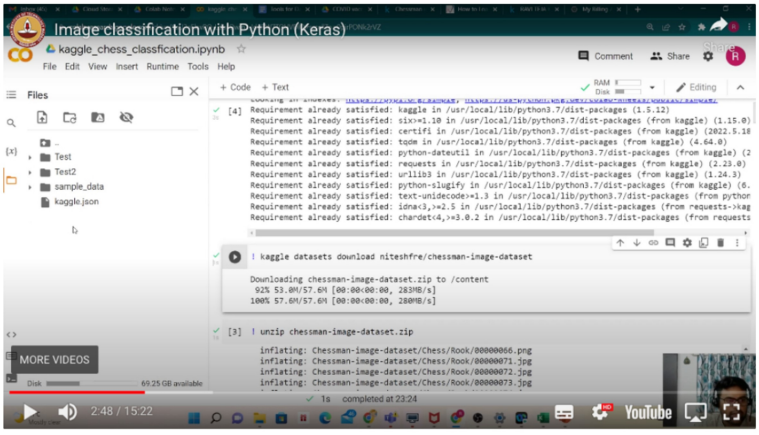


→ save as shape file /KML file, name it and save it in location of your choice

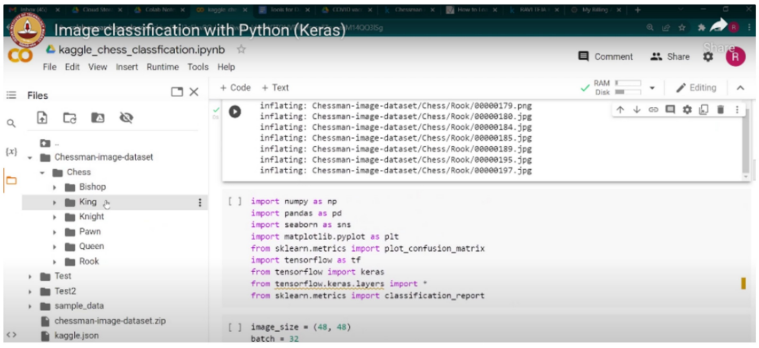
L2 Image classification with Python (Keras)

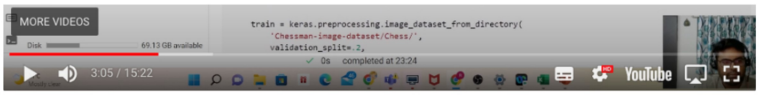
Image classification applications- finger print to unlock mobile, identify vehicles at traffic lights,..

1. install kaggle library
2. provide API JSON file into the google collab - go to account details on kaggle→ scroll down to API → create new API token → corresponding JSON file with user your ID and credentials gets downloaded → upload the file into the directory
3. now we can directly import the kaggle dataset into our working environment (using given code)



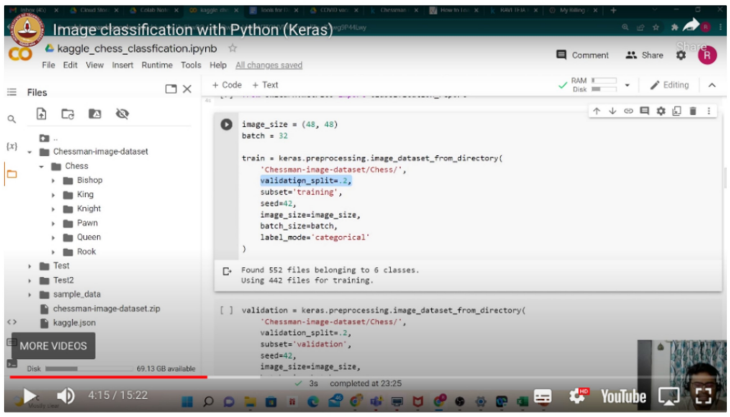
4. unzip the file (using given code)





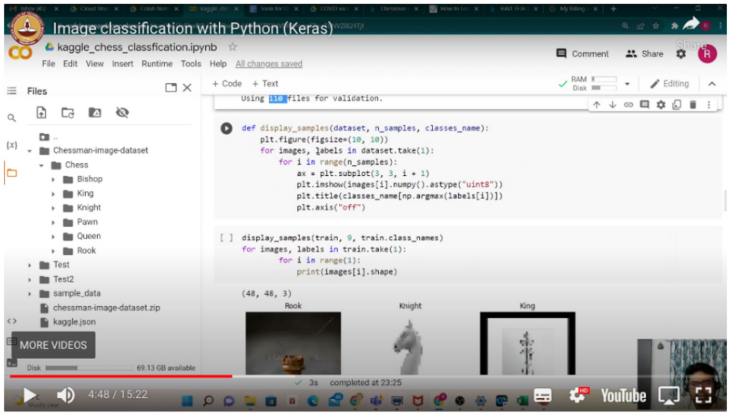
after refreshing, corresponding 6 classes for chess dataset are available

5. Building a neural network that builds on the dataset to classify the images to one of these 6 classes (multi-class image classification problem)
    - import necessary libraries
    - split the data into validation and train → keras comes with an inbuilt function "preprocessing" where we specify the ratio of the validation and training split



(here 0.2 refers to 20% validation, 80% training)

- displaying sample images



- looking at size of the labels we have in the training data set

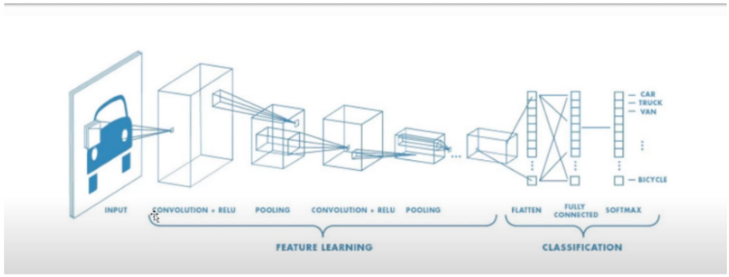
```
class_names = train.class_names
labels = np.array([])
for _, label in train:
    labels = np.concatenate((labels, np.argmax(label, axis=-1)))
_, counts = np.unique(labels, return_counts=True)
```

- image classification neural network

```
input_shape = (image_size[0], image_size[1], 3)
reg = keras.regularizers.l2(0.0005)

model = keras.Sequential()
model.add(Conv2D(32, (3, 3), padding="same", activation="relu", input_shape=image_size + (3,), kernel_regularizer=reg))
model.add(MaxPooling2D(pool_size=(2, 2)))


model.add(Conv2D(64, (3, 3), padding="same", activation="relu", kernel_regularizer=reg))
model.add(MaxPooling2D(pool_size=(2, 2)))


model.add(Conv2D(128, (3, 3), padding="same", activation="relu", kernel_regularizer=reg))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(len(train.class_names), activation='softmax'))


model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 48, 48, 32)        896

 max_pooling2d (MaxPooling2D  (None, 24, 24, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 24, 24, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 12, 12, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 12, 12, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 6, 6, 128)        0
 2D)

 dropout (Dropout)           (None, 6, 6, 128)         0

 flatten (Flatten)           (None, 4608)              0

 dense (Dense)               (None, 128)               589952

 batch_normalization (BatchN  (None, 128)              512
 ormalization)

 dropout_1 (Dropout)         (None, 128)               0

 dense_1 (Dense)             (None, 6)                 774

=================================================================
Total params: 684,486
Trainable params: 684,230
Non-trainable params: 256
```

what a neural network essentially is:



max pooling- for every 2 x 2 value it chooses the max from that

i.e.

Max Pooling

0	49	0	35
0	80	33	1
30	98	1	75
0	10	1	95

80	35
98	95

- specify whether single or multi class classification problem

```
[ ]  1 model.compile(
     2     loss="categorical_crossentropy",
     3     optimizer="adam",
     4     metrics=["accuracy"]
     5 )
     6
     7 epochs = 8
     8 model.fit(
     9     train,
    10     epochs=epochs,
    11     validation_data=validation
    12 );
```

```
Epoch 1/8
14/14 [==============================] - 7s 282ms/step - loss: 2.4486 - accuracy: 0.1900 - val_loss: 5.4763 - val_accuracy: 0.1091
Epoch 2/8
14/14 [==============================] - 6s 260ms/step - loss: 2.1721 - accuracy: 0.1923 - val_loss: 3.8731 - val_accuracy: 0.2636
Epoch 3/8
14/14 [==============================] - 6s 261ms/step - loss: 2.0739 - accuracy: 0.2195 - val_loss: 2.9415 - val_accuracy: 0.1545
Epoch 4/8
14/14 [==============================] - 6s 261ms/step - loss: 2.0636 - accuracy: 0.2285 - val_loss: 2.0378 - val_accuracy: 0.2727
Epoch 5/8
14/14 [==============================] - 6s 261ms/step - loss: 2.0147 - accuracy: 0.2217 - val_loss: 2.6708 - val_accuracy: 0.1273
Epoch 6/8
14/14 [==============================] - 6s 263ms/step - loss: 1.8297 - accuracy: 0.2692 - val_loss: 1.9581 - val_accuracy: 0.2455
Epoch 7/8
14/14 [==============================] - 6s 260ms/step - loss: 1.8315 - accuracy: 0.3213 - val_loss: 1.7600 - val_accuracy: 0.3364
Epoch 8/8
14/14 [==============================] - 6s 263ms/step - loss: 1.6807 - accuracy: 0.3575 - val_loss: 1.7911 - val_accuracy: 0.3182
```

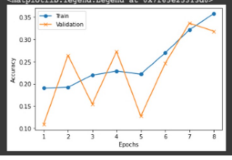
→specifying loss as “categorical cross entropy” which is a requirement when it’s a multi-class classification problem

→accuracy increasing on a step by step bases as the epochs increase

- plotting accuracy for training and validation dataset

```
[ ]  1 epochs_range = [i+1 for i in range(epochs)]
     2 plt.plot(epochs_range, model.history.history['accuracy'], '-o', label='Train')
     3 plt.plot(epochs_range, model.history.history['val_accuracy'], '-x',label='Validation')
     4
     5 plt.ylabel('Accuracy')
     6 plt.xlabel('Epochs')
     7
     8 plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f65e25313d0>
```



→ accuracy increasing w.r.t training dataset

→ accuracy fluctuating w.r.t validation dataset (reasons could be overfitting of the data, increase raw data sample, reduce convolution layers, etc)

- confusion matrix

```
[ ]  y_pred = np.argmax(model.predict(validation), axis=-1)

predictions = np.array([])
labels =  np.array([])
for x, y in validation:
    predictions = np.concatenate([predictions, np.argmax(model.predict(x), axis=-1)])
    labels = np.concatenate([labels, np.argmax(y.numpy(), axis=-1)])
```

```
[ ]  conf = tf.math.confusion_matrix(labels=labels, predictions=predictions)
     sns.heatmap(conf, annot=True, cmap='Blues', yticklabels=class_names, xticklabels=class_names)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f65e65b77d0>
```

tds_week6.5.docx

04 August 2022 14:28

6.5 : Visualising network data with kumu

- **Kumu :**
  - **A tool that helps visualize complex relationships that access data**
  - **Basically helps understand relationship between different entities**
  - Eg. We can find out the common interests of people in different communities (as part of a social network analysis for instance)

Example : trying to find actors who worked together on movies (Indian only)

Aim: to get a matrix where the rows and columns denote the actors and the values in the matrix are 1 if the two actors have worked together and 0 if they have not worked together . Thus for eg if there are 10 actors there will be 10 rows and columns)

Matrix

Movies / Actors	A1	A2	A3	A4
M1	1	0	1	1
M2	1	1	0	0
M3	0	0	0	1

Matrix Transposed

	M1	M2	M3
A1	1	1	0
A2	0	1	0
A3	1	0	0
A4	1	0	1

A4	1	0	1	

(multiplying this matrix and its transpose to get the desired matrix)

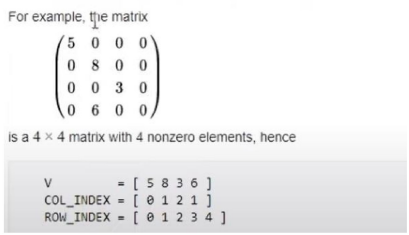
Matrix Transposed * Matrix

	A1	A2	A3	A4
A1	2	1	1	1
A2	1	1	0	0
A3	1	0	1	1
A4	1	0	1	2

So the diagonal elements in the matrix denote the number of movies that particular actor has acted in eg A1 : 2 implying that A1 has acted in two movies totally. The rest of the elements denote the number of movies the two actors have acted in together

However such a matrix will be very sparse. (there will be a lot of zero elements)
Hence we use the compressed sparse row

- **Compressed Sparse Row**



In the above matrix :

- V denotes all the non zero elements
- Col_index is the column index of all the non zero elements
- Row_index is the number of non zero elements above current row j. eg. If j=1, then 0 non zero elements above it, if j=2, 2 non zero elements above row j. The last element in the row_index denotes the number of non zero elements.
- Thus, a 16 element matrix can be expressed with just 13 elements. This has

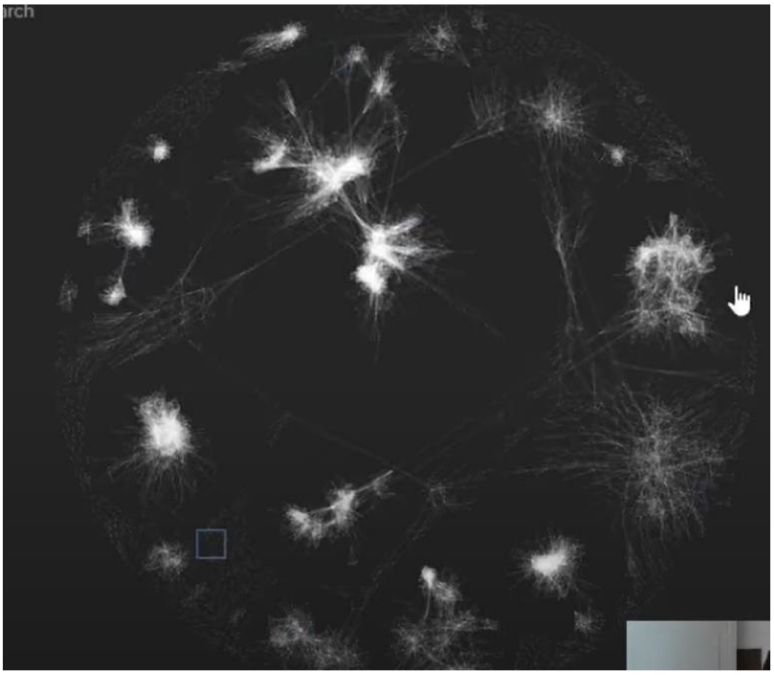
row j. The last element in the row_index denotes the number of non zero elements.

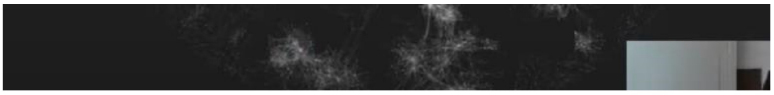
- Thus, a 16 element matrix can be expressed with just 13 elements. This has consequential effect while scaling if the number of non zero elements is less while compared to total number of elements
  - **Python scipy library helps form this csr matrix**

```
from scipy.sparse import csr_matrix
```

Once the required dataframe is uploaded into kumu
We get a network like this

Clusters and links can be seen between clusters

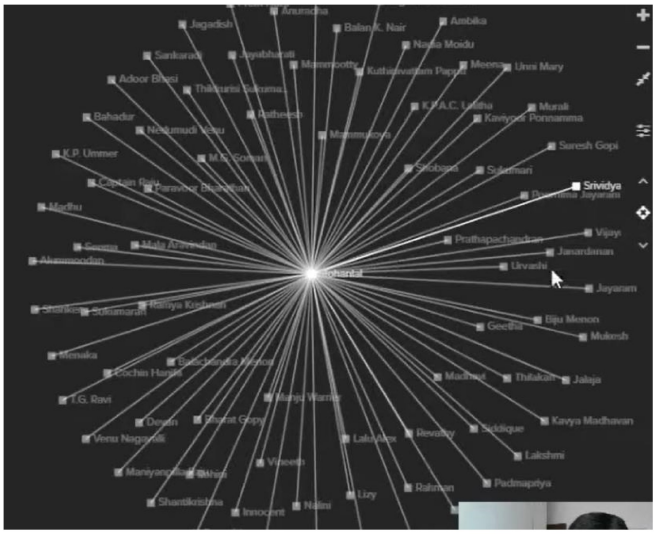




Eg. A quick search for actor Mohanlal pulls up this particular cluster, showing all the actors he's acted with (his direct connections)

Steps:

- Click on the actor
- Focus
- direct



And clicking on a specific actor Mohanlal has acted with gives the count of the number of movies they've done together

Mohanlal and Saikumar

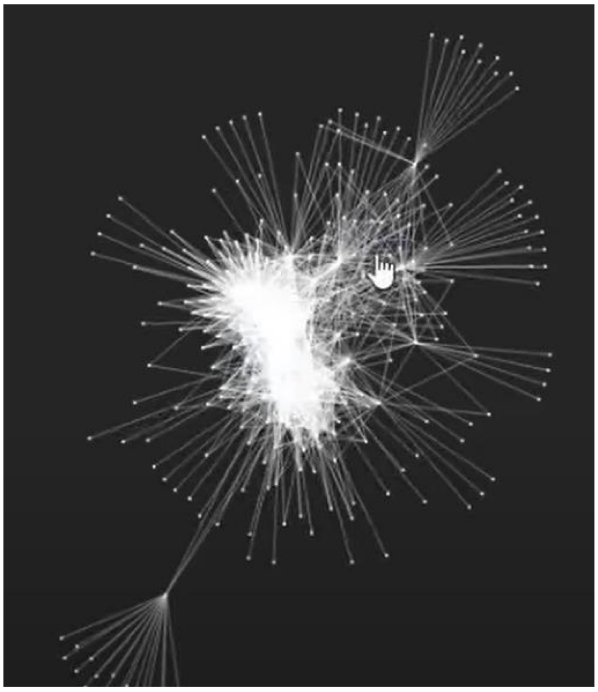
COUNT 8

Mohanlal's indirect connections :

Steps:

- Click on the actor
- Focus
- indirect

- Click on the actor
- Focus
- indirect



6.4 6.5

04 August 2022 14:32

6.4 VISUALIZING ANIMATED DATA WITH FLOURISH

- Animation and Flourish used to both engage and inform.
- Engage movement generally draws audience and generate excitement
- It also helps to inform to help explain the data
- Object Consistency: It is a graphical element that represents a particular data point that can be tracked visually through the transition.
- The advantage of object consistency is that it lessens the burden by preprocessing of the motion rather than scanning it sequentially.
- Animation is a integral part of template for its visualization.
- Templates can also work well, standalone on the social media where interactive web content is not necessary to deliver the value of visualization.
- Animation Setting in Line Chart:
    a. For the animation in the Line Chart to start off there are just two settings animation duration and mode duration.
    b. The settings in the Line Charts are in milliseconds.
    c. Stage: The movement between one data point and the next.
    d. Mode duration: This is the transition that runs when we switch between two different views, scores and ranks.
- Animation Setting in Bar Chart:

    a. For the animation in the Bar Chart we use timeline settings which indicates the duration required to go through the data series rather than in stages.
    b. The other setting (bar rank animation duration) help us to control the animation speed of the bars moving positions.
- Line Bar Pie template animates in terms of both drawing and morphing.
- Both the elements of the LBP template is controlled by Animation duration.
- We need to change the animation setting down to zero if we want to change the sort of morphing in the story along with animation.
- Animation Settings in LineBar Pie Chart
    a. Only animate with the same name: This is the default setting in the chart ,when switched on flourish only attempt to series animate with the same name i.e the same column header.
    b. When switched them flourish also attempt to animate the series with the different name.
    c. ON setting is the default setting as the series with different name can cause the misleading effect.
- In Scatter template, in order to animate the template needs to know what to animate based upon.
- First step is to set the name so that flourish knows what to animate.
- Animation Settings in the Scatter Chart

    a. Animation Duration : duration of animations in seconds.
    b. Animation Stagger: the delay between each dot starting to move
- Survey has two animation settings Animation duration and Animation Stagger.(in milliseconds)
- Heat Map has two settings Fade and Flip animation settings (in seconds).
- Spider or radar has Animation duration which is both for drawing and morphing animations as in line bar pie chart. (in seconds)
- In hierarchy, there is animation duration which is just a speed setting in seconds.
- Data Explorer template is quite powerful and combine the elements of the survey, scatter, hierarchy and projection maps.
- Allows to morphs from dots to maps.

6.5 VISUALIZING NETWORK DATA WITH KUMU

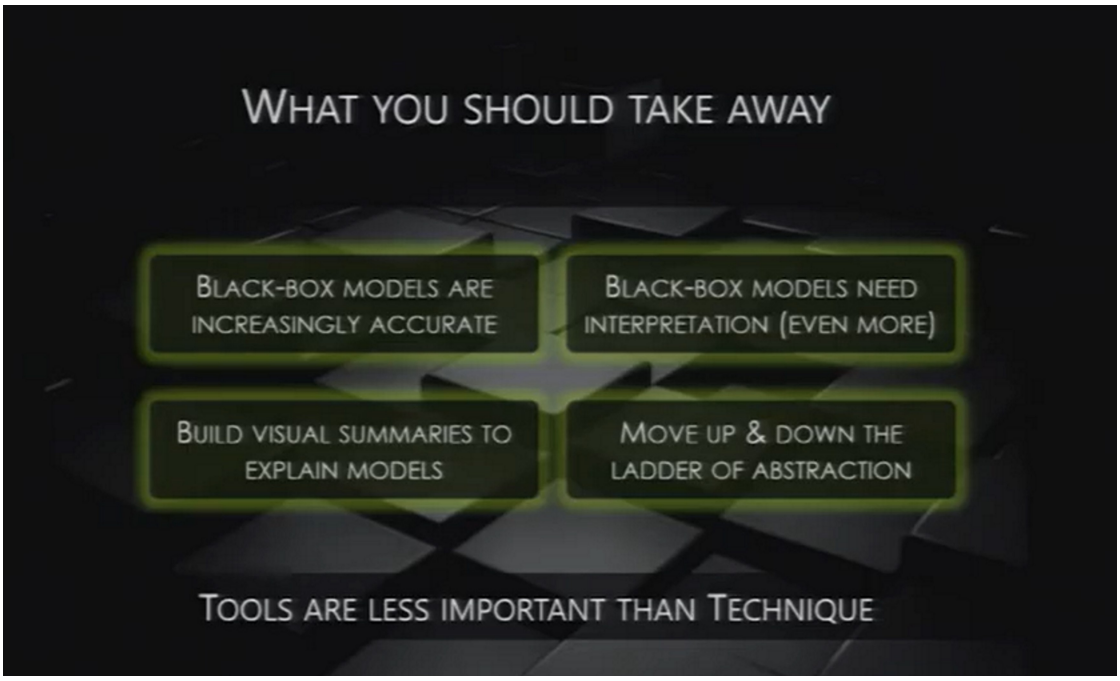
- Kumu is a tool that allows to visualize the complex relationship to access the data.

Module 6b

01 August 2022 21:53

What is black box model?

->produces useful information without revealing about its internal working.

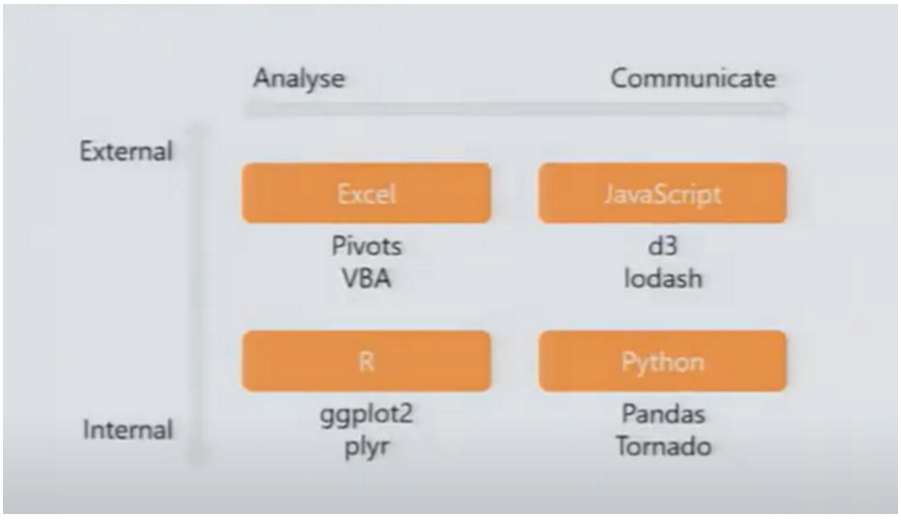


->

Being a good data storyteller is an art as well as a science. **Data Scientists take the help of various data visualization tools like Tableau to present the data in a visually appealing format**. A Data Scientist not only understands the data but also understands the business and the end user very well.

-> We will learn how to explain visually our models to the decision makers of business.

tools



General Purpose tool– Excel, Google Data Studio, Power BI, Tableau
Specialized purpose tool– Excel(VBA), Flourish Studio(for better animation), Kumu (network visualization), QGIS(geographic visualization)

Flourish

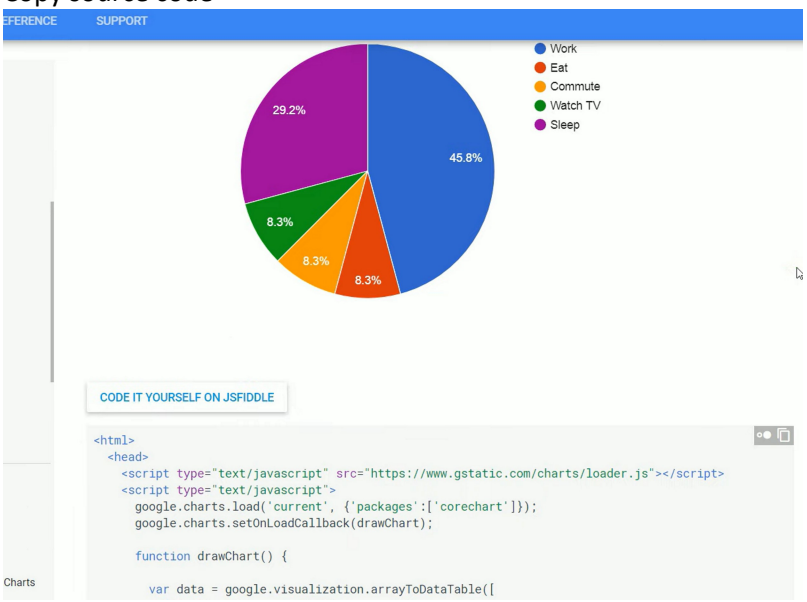
Website: flourish.studio
It helps to create high end visualizations without any coding.

Google Chart

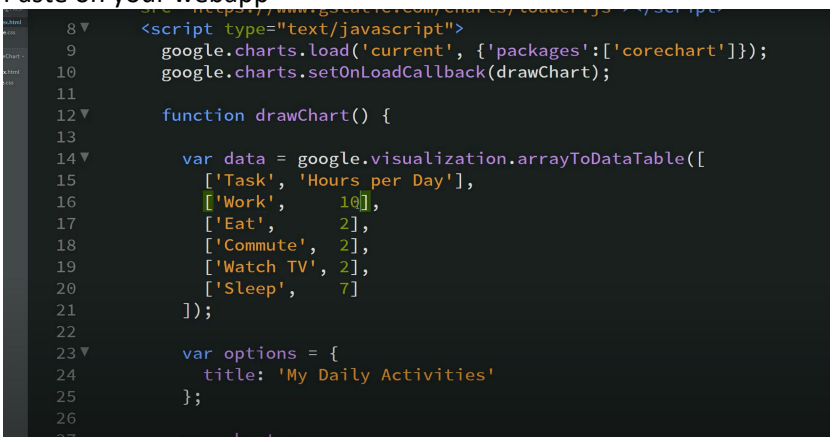
1. Extensive library of plots / charts
a. Useful to browse around for inspiration on how to tell a story using your data (e.g., Sankey / alluvial charts)

2. User supplies information; Google charts returns graphical charts

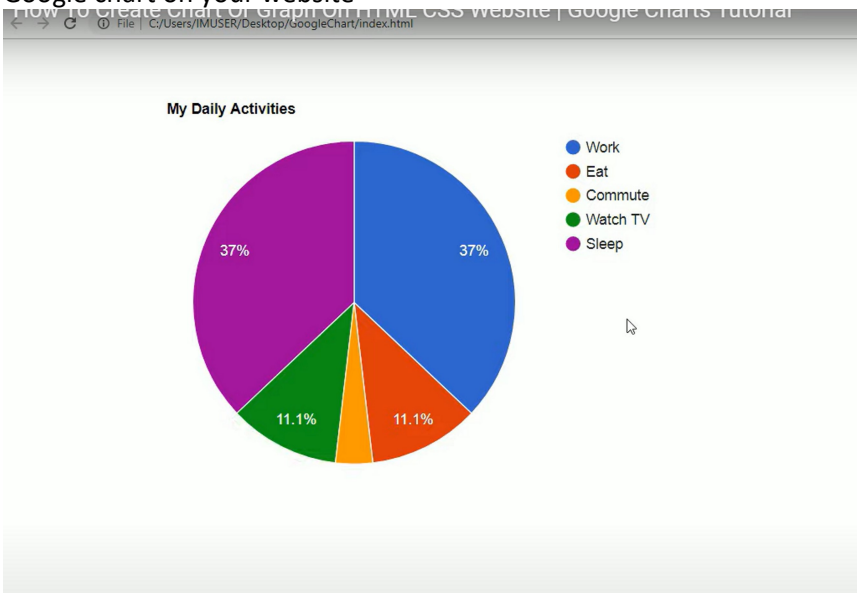
Copy source code



Paste on your webapp

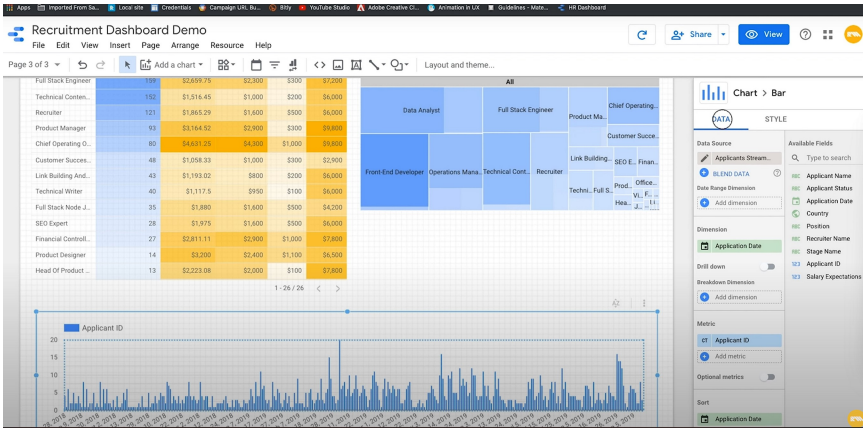


Google chart on your website



Google data studio

1. Create simple dashboards / reports using basic interactive charts
2. Easier to learn compared to Tableau / Power Bi
3. ... is free



Ideal for

1. Simple reports that require very little to no data processing / cleaning
2. High level reports that do not dive deep into finer details
3. When budget is a constraint

Four major concepts in google data studio:–

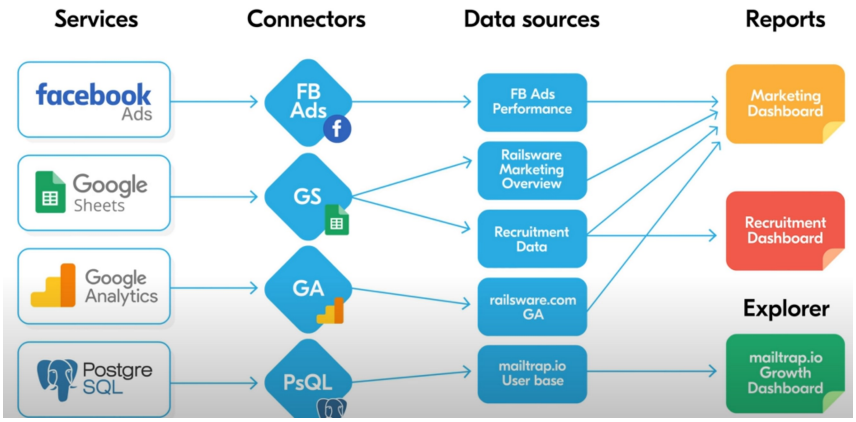
Connectors, Data Sources, Reports & Explorer

Data sources-Reports have many to many relationships

Connector is an interface of receiving data from various platforms.

Data source is a blueprint of data that can be modified

Explorer- to visualize data quickly



Some pandas formulas

read_csv()

read_csv() function helps read a comma-separated values (csv) file into a Pandas DataFrame

head()

head(n) is used to return the first n rows of a dataset.

loc[:]

loc[:] helps to access a group of rows and columns in a dataset, a slice of the dataset, as per our requirement

drop_duplicates()

drop_duplicates() returns a Pandas DataFrame with duplicate rows removed.

groupby()

groupby() is used to group a Pandas DataFrame by 1 or more columns, and perform some mathematical operation on it. groupby() can be used to summarize data in a simple manner.

merge()

merge() is used to merge 2 Pandas DataFrame objects or a DataFrame and a Series object on a common column (field).

sort_values()

sort_values() is used to sort column in a Pandas DataFrame (or a Pandas Series) by values in

ascending or descending order.

fillna()

Tfillna() helps to replace all NaN values in a DataFrame or Series by imputing these missing

values with more appropriate values.

```
data_1['City temp'].fillna(38.5, inplace=True)
```

DF[DF.category.isin({'actor','actress'})]
Take only those row where 'category' column is actor
Or actress

Value_counts

Return all unique value and its counts

Values

Return numpy representations of the given dataframe

Lambda function

DF.apply(lambda x: .....)

Data to excel

DF.to_excel('data.xlsx')

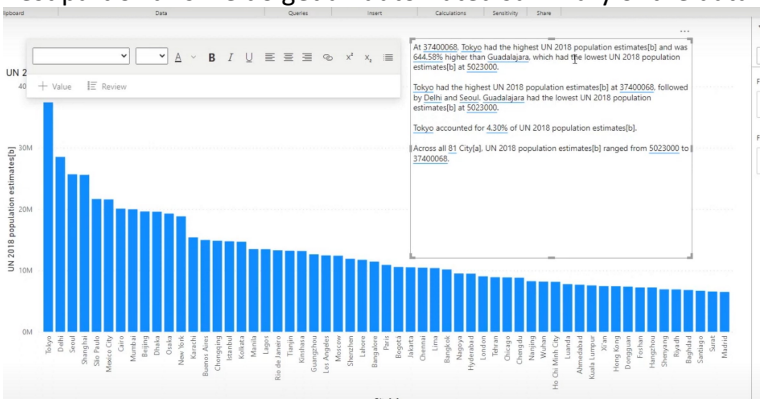
Module 7

01 August 2022 22:30

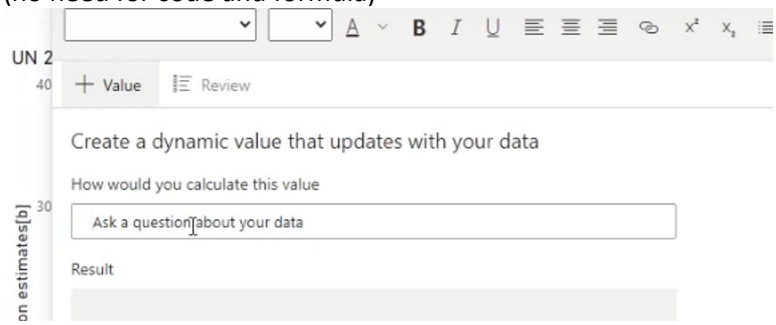
Narrate a story

Smart narratives with Power Bi

Power Bi by Microsoft
Visualizing the data
Provide insight from the underlying the data
Best part of bi is we do get an automated summary of the data

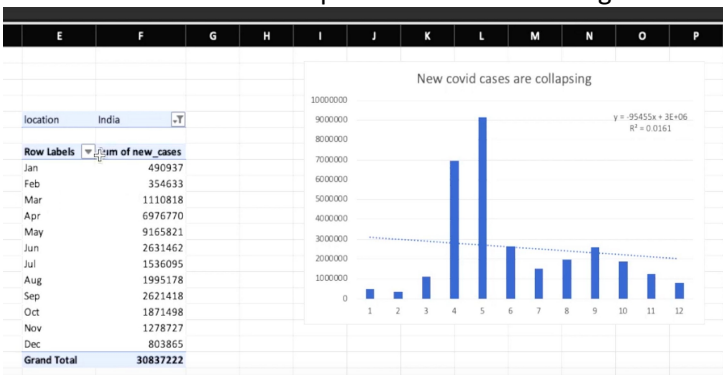


AI chatbot like feature to get information what we asked for by typing out question (no need for code and formula)

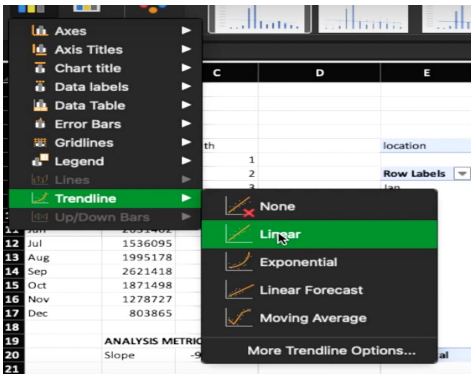


Narrative with Excel

To narrate that if data has positive trendline or negative.



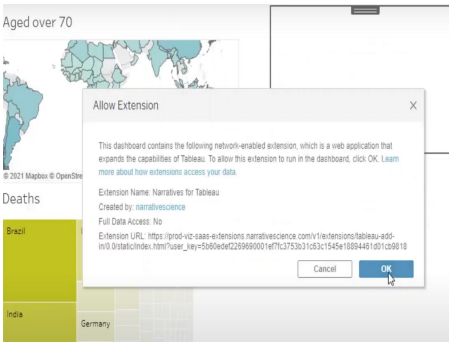
Add chart element->Trendline->Linear

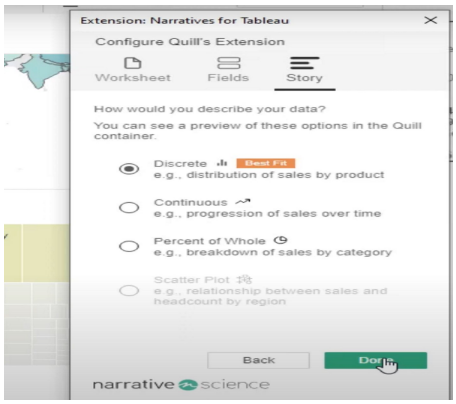


We can have linear regression predicted.
We get trendline visualization

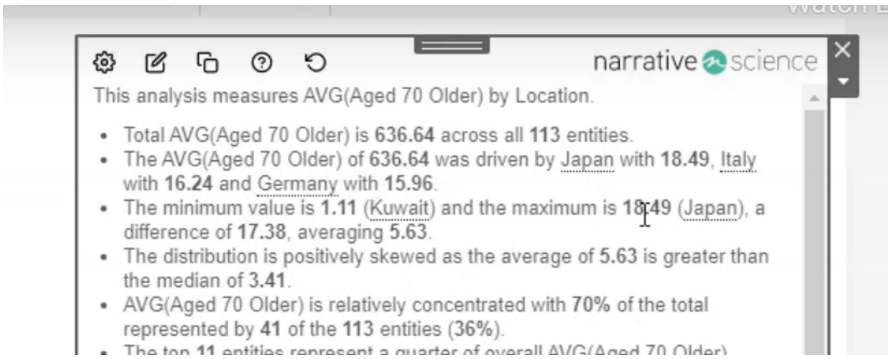
Narrate on Tableau with Quill

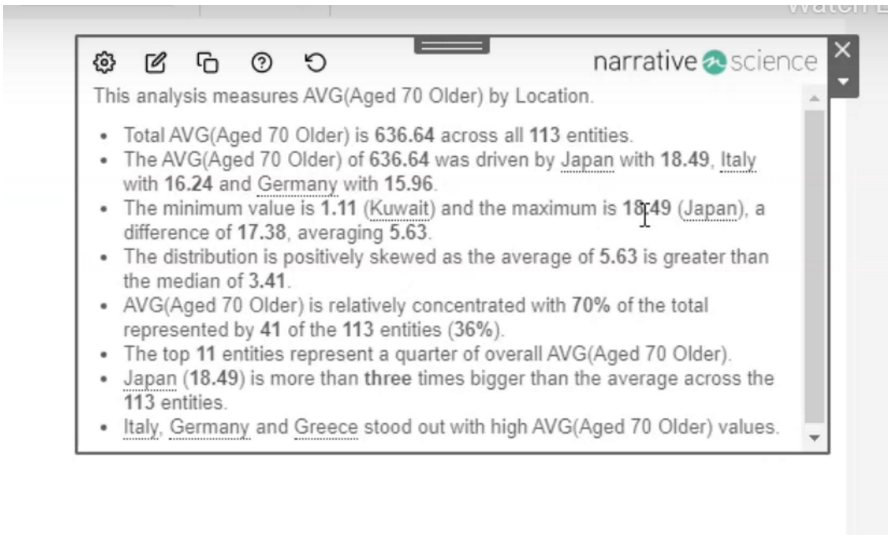
Quill is an extension that integrates with tableau dashboard





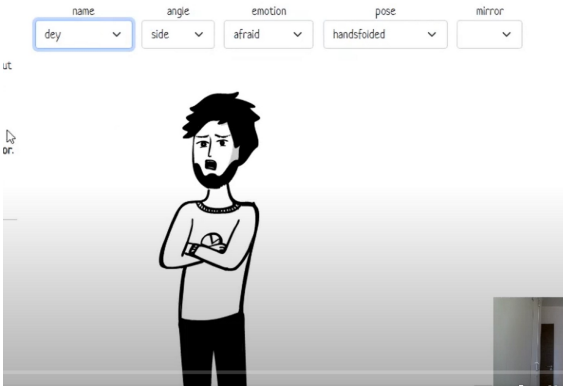
We can have many meaningful narrations which automatic generated by Quill



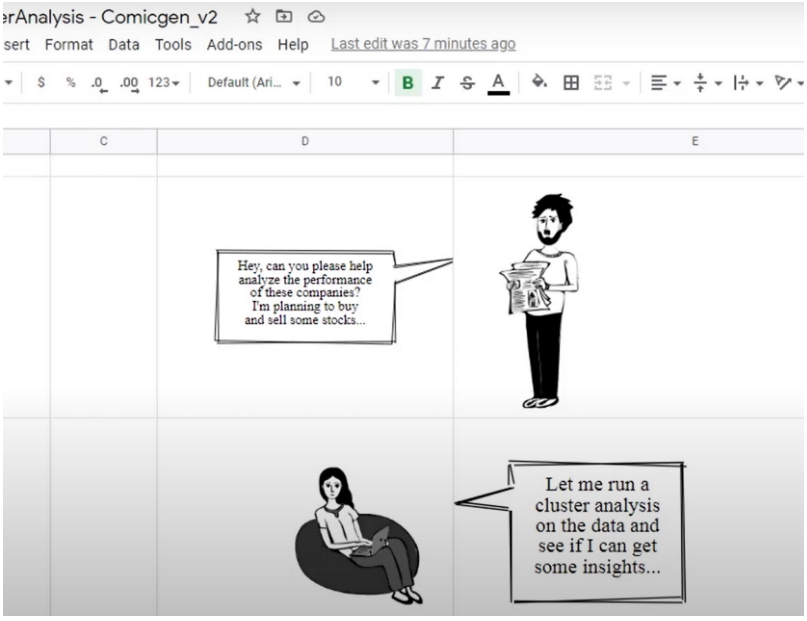


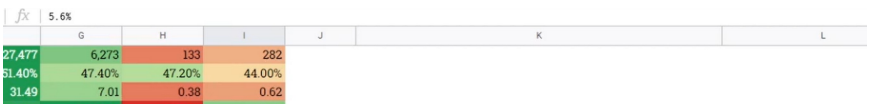
Comic narratives with Google sheets & comicgen

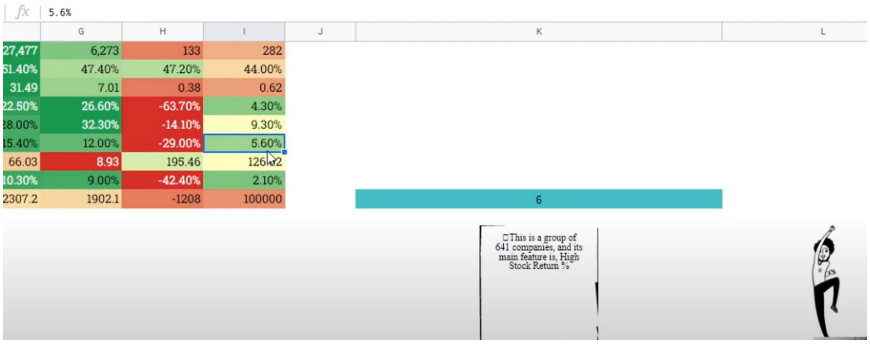
Comicgen allows us to create comics.



It helps us tell stories of our data.






fx
5.6%
G
H
I
J
K
L
27,477
6,273
133
282
51.40%
47.40%
47.20%
44.00%
31.49
7.01
0.38
0.62
22.50%
26.60%
-63.70%
4.30%
28.00%
32.30%
-14.10%
9.30%
15.40%
12.00%
-29.00%
5.60%
66.03
8.93
195.46
126.42
10.30%
9.00%
-42.40%
2.10%
2307.2
1902.1
-1208
100000
6
This is a group of 641 companies, and its main feature is, High Stock Return %s