iNeuron

# Low Level Design

NBA Draft Combine Measurements

Swetanshu Pandey

| | |
|---|---|
| Written by | Author 1 |
| Document Version | 0.1 |
| Last Revised Date | |

1. NBA Draft Combine Measurement

# DOCUMENT CONTROL

Change Record:

| Version | Date | Author | Comments |
|---|---|---|---|
| 0.1 | 10-Apr-2022 | Author 1 | Introduction |
| | | | |
| | | | |

## Approval

| Version | Review Date | Reviewed By | Approved By | Comments |
|---|---|---|---|---|
| | | | | |

# Contents

# 1. Introduction

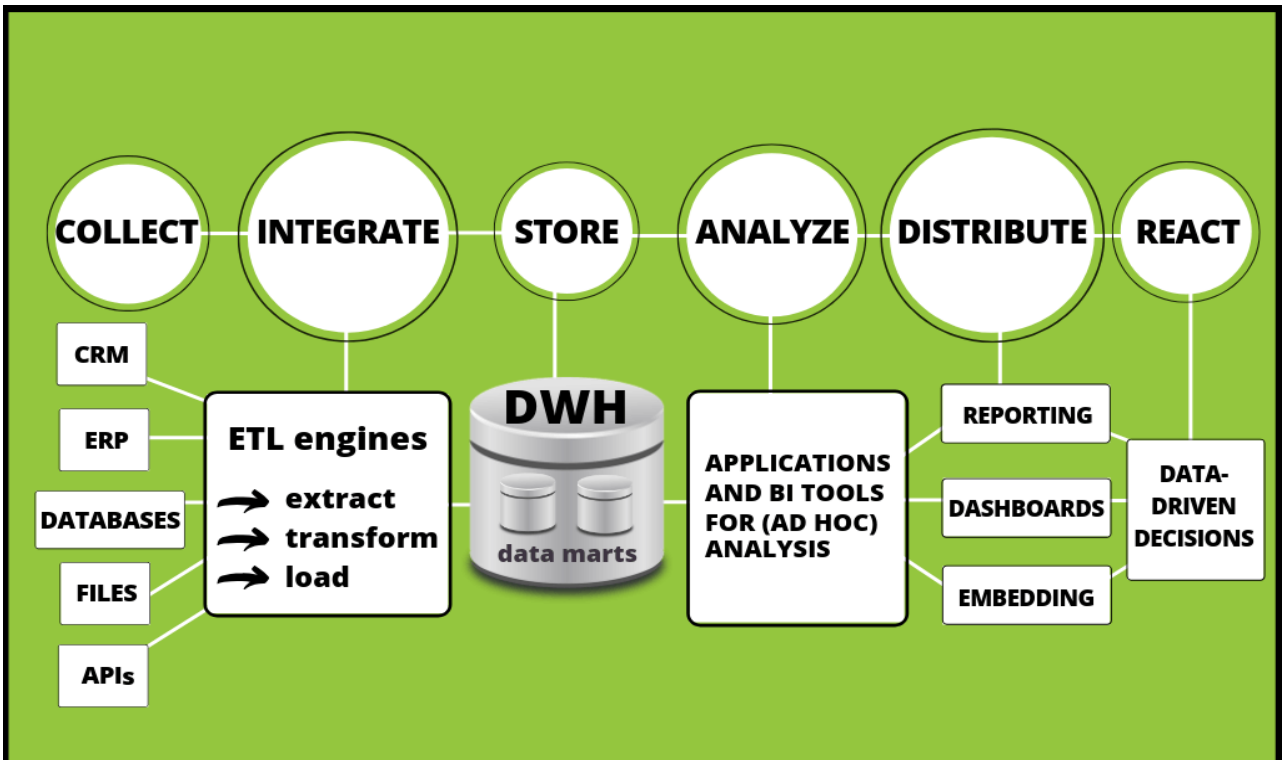
## 1.1 What is Low-Level design document?

The goal of the LDD or Low-level design document (LLDD) is to give the internal logic design of the actual program code for the House Price Prediction dashboard. LDD describes the class diagrams with the methods and relations between classes and programs specs. It describes the modules so that the programmer can directly code the program from the document.

## 1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

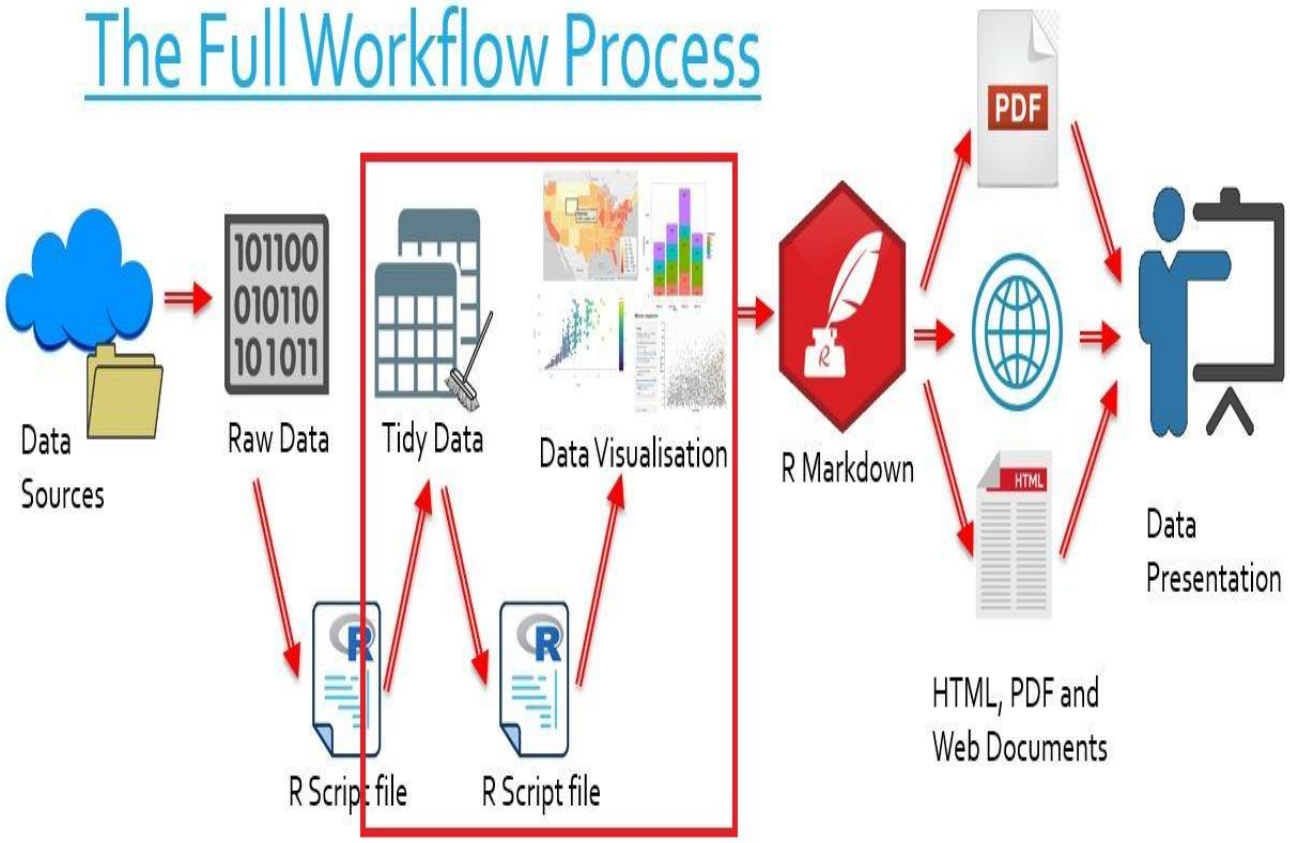
## 2. Architecture




## Plotly Server Architecture

Almost every Data Science project requires some kind of visualization, like visualizing the input data, exploratory data analysis using histograms or scatter plots, finding outliers or plotting statistics using box and whisker plots, visualizing the relationship between nodes using network diagrams, checking the relationships between variables using correlation matrices, visualization techniques to help understand relationships within high-dimensional datasets, visualizing the performance of the models, or the train history, etc.

## Plotly Communication Flow




# 3. Architecture Description

## 3.1. Data Description

The data set contains weight, height ,sprint , body fat , player name,wingspan,agility,etc. of the NBA players.

1.Year – In which year player played the match (in integer).

2.Body Fat - player's fat on his body (in float)

6 .NBA Draft Combine Measurement



3.Wingspan - player's wingspan (in float)

4. Sprint - player's sprint value (in float)

5. Bench - player's bench value (in float)

6. Standing Reach - player's standing reach value (in float)

7.Agility - player's agility value (in float)

8. Draft pick – player's draft pick value (in float)

9.Weight –Player's weight (in float)

## 3.2. Web Scrapping

Web scraping is a technique to automatically extract content and data from websites using bots. It is also known as web data extraction or web harvesting. Web scrapping is made simple now days, many tools are used for web scrapping. Some of python libraries used for web scrapping are Beautiful Soup, Scrapy, Selenium, etc.

## 3.3. Data Transformation

In the Transformation Process, we will convert our original datasets with other necessary attributes format. And will merge it with the Scrapped dataset.

## 3.4. Data Insertion from csv files

1 . Download the dataset from open sources(kaggle) or github and store it in your local system where you can easily access it.

2 Import important libraries required for viewing the dataset in your local IDE

## 3.5 Representation of results using python libraries

7 .NBA Draft Combine Measurement

8 LOW LEVEL DESIGN

iNeuron

### Step 1. Configuring Pandas, Numpy, Matplotlib, Seaborn

Launch jupyter on your local system and import the libraries as shown in the picture.

*Importing important libraries*

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

### Step 2. Import plotly and cufflinks

***Using Plotly***

```
from plotly.offline import iplot
import plotly as py
import plotly.tools as tls
import cufflinks as cf
```

```
import plotly.offline as pyo
import plotly.graph_objs as go
pyo.init_notebook_mode()
```

```
cf.go_offline()
```

8 .NBA Draft Combine Measurement

## 9 LOW LEVEL DESIGN


You need to initiate the Plotly Notebook with **init_notebook_mode** to use plotly in local environment**,** also note that when you call py.iplot it is still calling the plot function from online plotly module, you need to import the **iplot**(not plot) from **plotly.offline** and use it for offline plot and inside notebook rendering.

The cufflinks library binds the power of plotly with the flexibility of pandas for easy plotting cufflinks.

## Plots using cufflinks

```
[ ]: ### All cufflinks themes
```

```
[ ]: cf.getThemes()
```

```
[ ]: cf.set_config_file(theme='solar')
     df6.iplot(x = 'Wingspan' , y ='Player',mode = 'markers',color="pink", xTitle = "Year" , yTitle = 'Player')
```

```
[ ]: # The above plot shows that most of the players have wingspan between 80-86 and
     # the players having maximum wingspan is Rudy Gobert
```

```
[ ]: df6.plot(figsize=(20,20))
```

```
[ ]: cf.set_config_file(theme = 'pearl')
     df6.iplot(kind ='scatter',mode = 'markers',x = 'Sprint',y = 'Body Fat',xTitle='Sprint',yTitle = 'Year')
```

**Step 3. Import plotly express**

# import PLOTLY.EXPRESS ¶

```
import plotly.express as px
```

To use the plotly express we need to first import plotly.express and the we can call different graphs of plotly.express.

***Graphical plots using plotly express***

```
fig = px.parallel_categories(df6,color='Year',color_continuous_scale=px.colors.sequential.Inferno)
fig.show()
```

```
fig = px.line(df6,x="Year",y="Weight",color = "Sprint",line_group="Player",hover_name="Player",line_shape="spline",render_mode="
fig.show()
```

***Area plot betwen year and player***

```
fig = px.area(df6,x="Year",y="Player",color = "Draft pick",line_group="Hand (Width)")
fig.show()
```

```
###### Sunburst plot
fig = px.sunburst(df6,path=['Standing reach','Body Fat'],values='Agility',color = 'Hand (Length)')
fig.show()
```

10 .NBA Draft Combine Measurement

### Step 4: Configuring Data Source

***Importing all dataframes***

```
[3]: df1=pd.read_csv("2012_nba_draft_combine.csv")
     df2=pd.read_csv("2013_nba_draft_combine.csv")
     df3=pd.read_csv("2014_nba_draft_combine.csv")
     df4=pd.read_csv("2015_nba_draft_combine.csv")
     df5=pd.read_csv("2016_nba_draft_combine.csv")
```

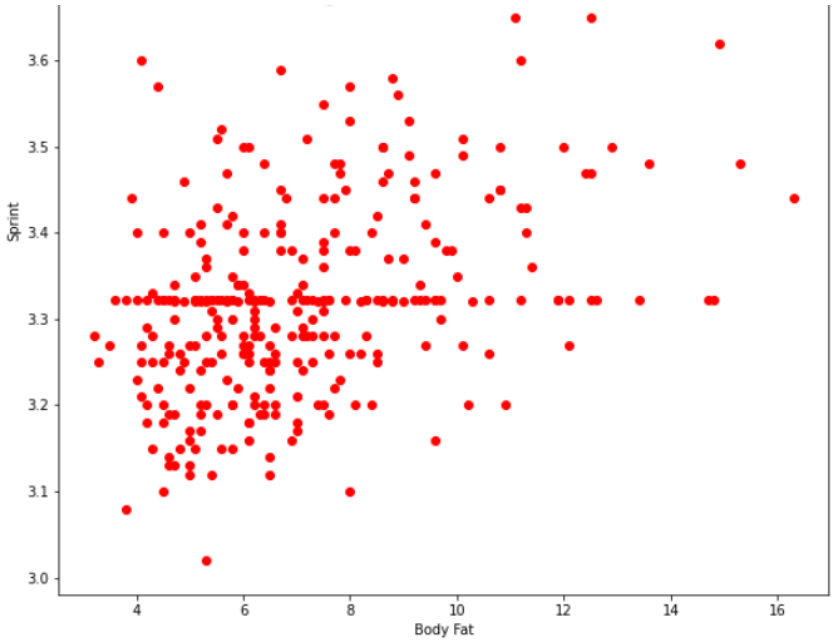
***Combining all dataframes***

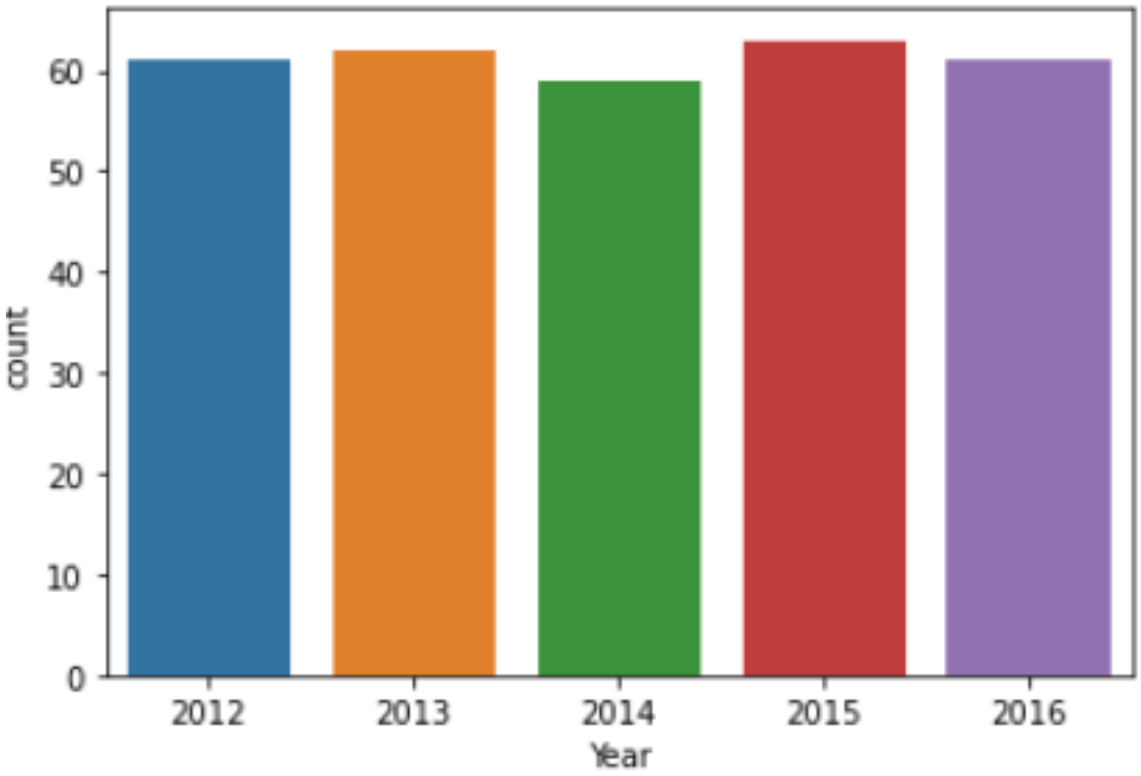
```
[4]: df6=pd.concat([df1,df2,df3,df4,df5],axis=0)
```

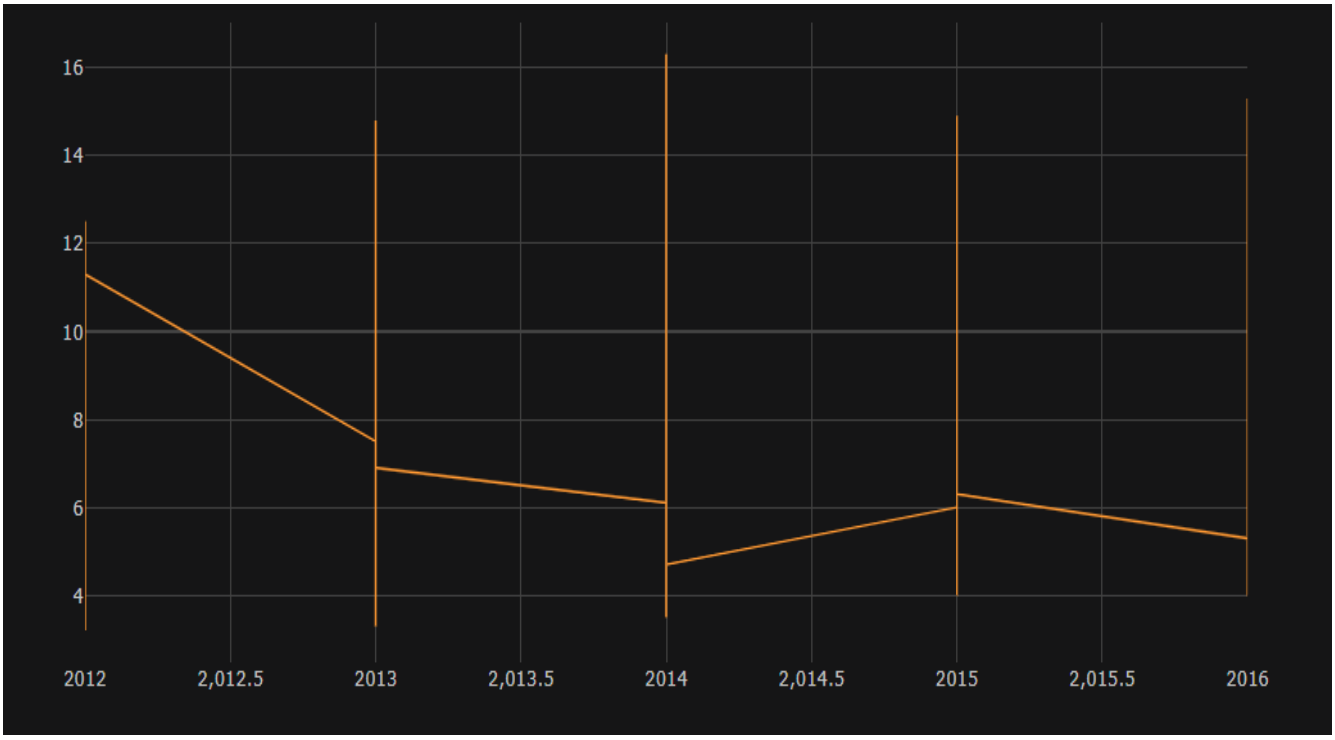
The data can be found from open source and can be imported and merged using pandas.

### Step 5: Deployment

Once we have completed all the coding part ,then its time to deploy our model and check the result.

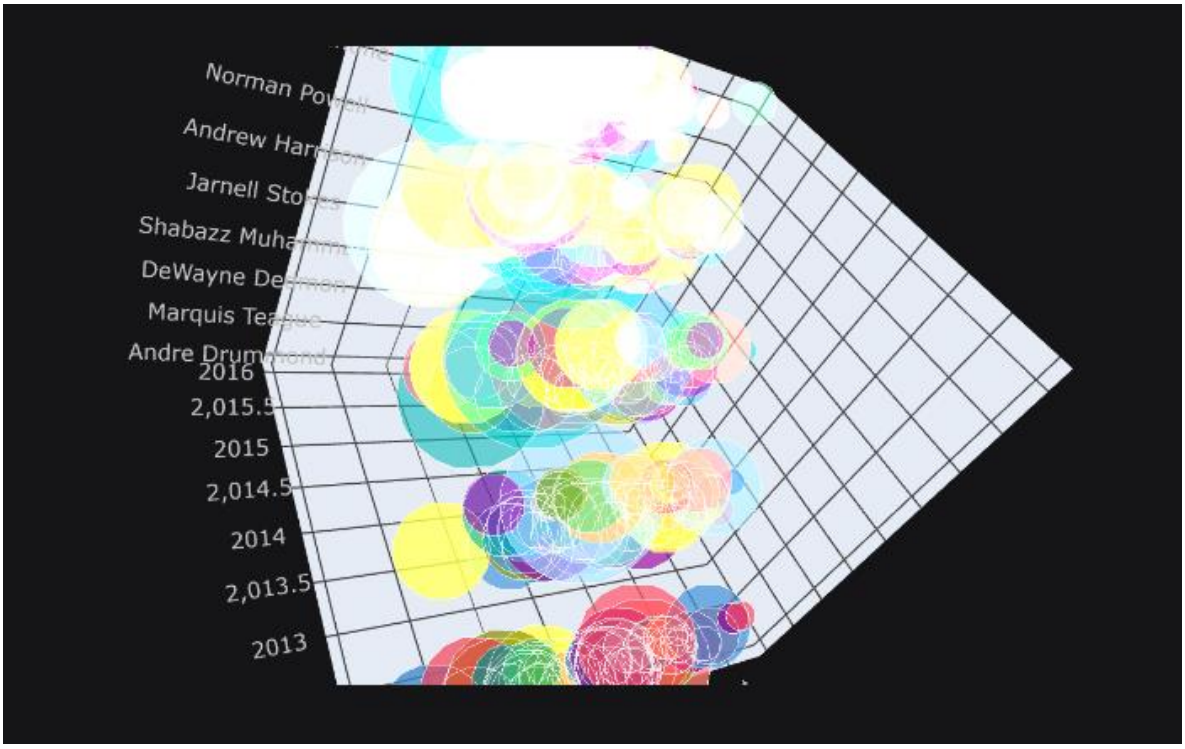


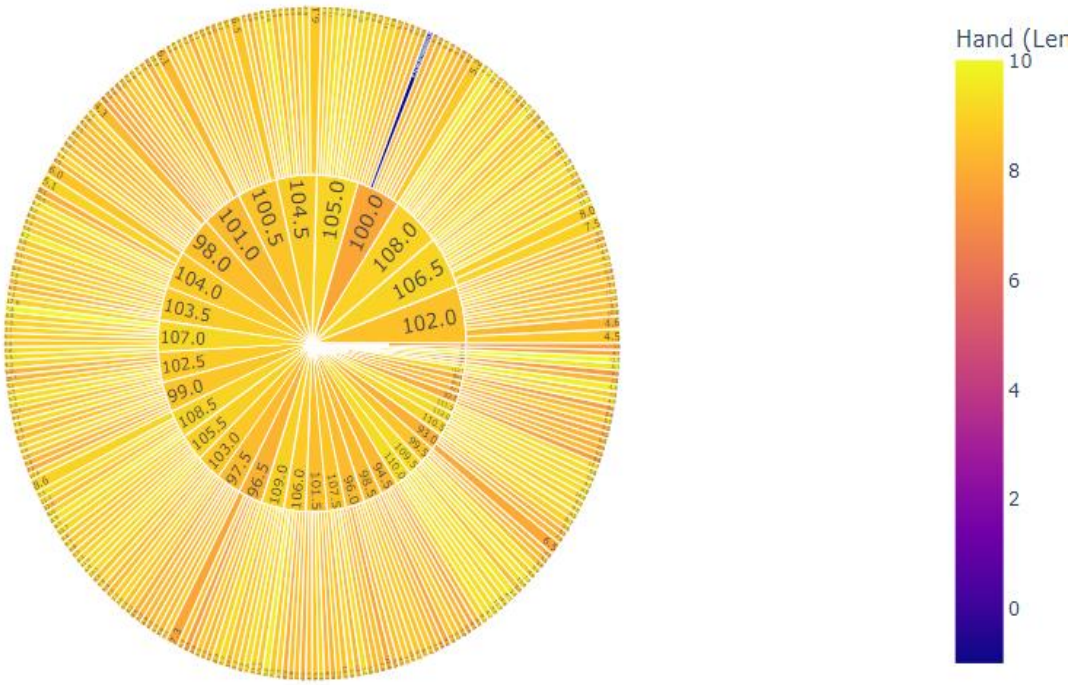

60
50
40
30
20
10
0
count
2012
2013
2014
2015
2016
Year


16
14
12
10
8
6
4
2012
2,012.5
2013
2,013.5
2014
2,014.5
2015
2,015.5
2016

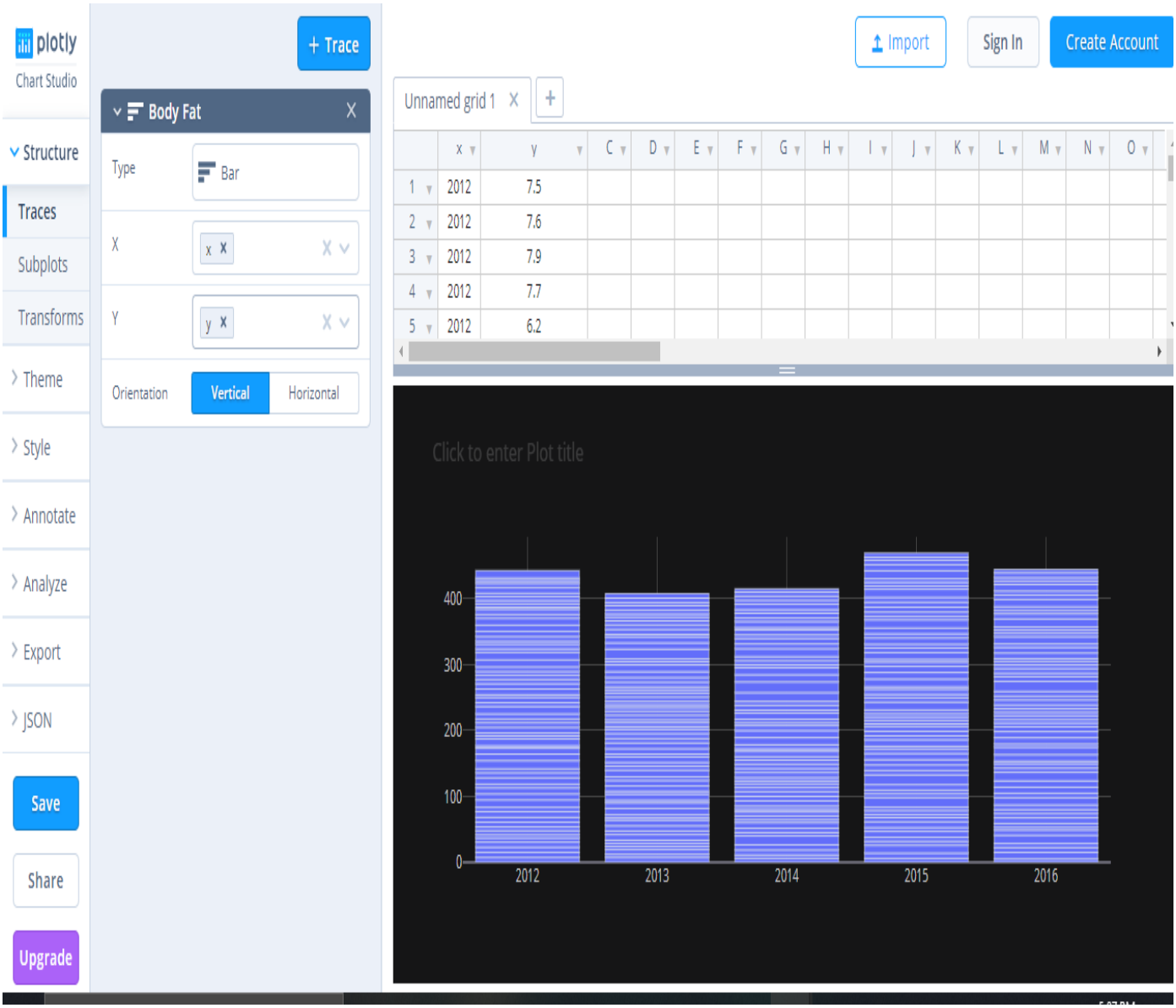


iNeuron


Norman Po
Andrew Har
Jarnell Sto
Shabazz Muha
DeWayne De
Marquis Tea
Andre Drum
2016
2,015.5
2015
2,014.5
2014
2,013.5
2013


105.0
104.5
100.5
101.0
100.0
98.0
108.0
104.0
106.5
103.5
102.0
107.0
102.5
99.0
108.5
105.5
103.0
97.5
96.5
109.0
106.0
101.5
101.5
99.0
98.5
94.5
110.0
93.0
99.5
109.5
Hand (Ler
10
8
6
4
2
0

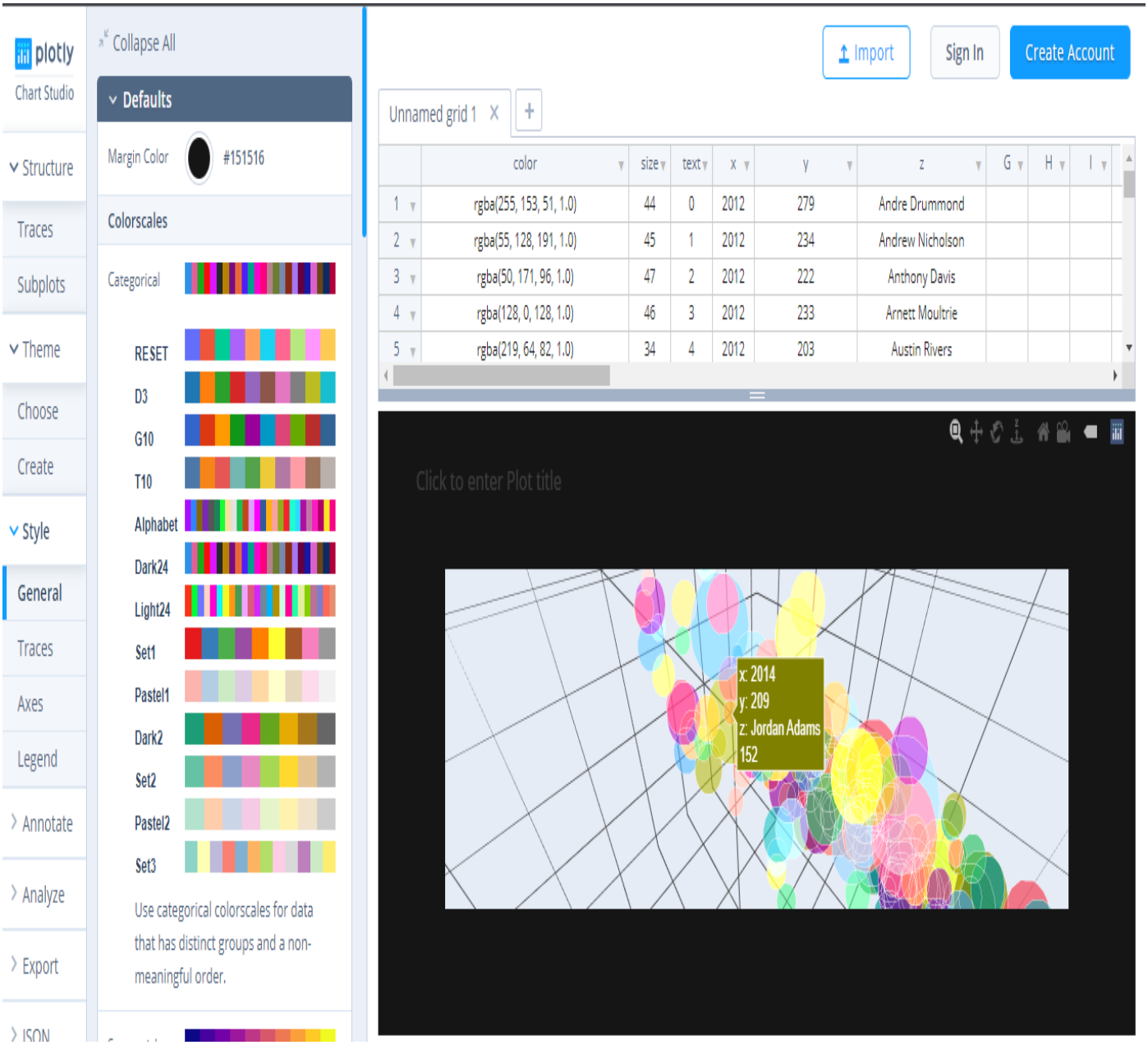
14.




| | x | y |
|---|---|---|
| 1 | 2012 | 7.5 |
| 2 | 2012 | 7.6 |
| 3 | 2012 | 7.9 |
| 4 | 2012 | 7.7 |
| 5 | 2012 | 6.2 |

LOW LEVEL DESIGN

15.


plotly
Chart Studio
Collapse All
Defaults
Margin Color
#151516
Colorscales
Categorical
RESET
D3
G10
T10
Alphabet
Dark24
Light24
Set1
Pastel1
Dark2
Set2
Pastel2
Set3
Use categorical colorscales for data that has distinct groups and a non-meaningful order.
Structure
Traces
Subplots
Theme
Choose
Create
Style
General
Traces
Axes
Legend
Annotate
Analyze
Export
JSON
Import
Sign In
Create Account
Unnamed grid 1
color
size
text
x
y
z
G
H
I
1 rgba(255, 153, 51, 1.0) 44 0 2012 279 Andre Drummond
2 rgba(55, 128, 191, 1.0) 45 1 2012 234 Andrew Nicholson
3 rgba(50, 171, 96, 1.0) 47 2 2012 222 Anthony Davis
4 rgba(128, 0, 128, 1.0) 46 3 2012 233 Arnett Moultrie
5 rgba(219, 64, 82, 1.0) 34 4 2012 203 Austin Rivers
Click to enter Plot title
x: 2014
y: 209
z: Jordan Adams
152