AMA With a Data Scientist

Ever wondered how and many other leading Data Scientists start their Data Science and Machine Learning Journey? Which all tools and techniques do they use in their day to day work, In which all domains does their expertise lay in? and many such questions. Many students and aspiring Data Scientists keep asking many Data Scientists about how to get started in DS/ML. I myself have personally asked many people about the same. This motivated me to create this notebook, where we will ask some basic questions every aspiring Data Scientist asks and let the data speak and give the optimal answers. As the questions are intended for Data Scientists,we will only check the survey responses by the Data Scientists in this notebook.

I hope this notebook becomes a good headstart for the aspiring Data Scientists. If you find this notebook useful and interesting, Do Upvote. Suggestions and feedback are always welcome and appreciated!!

In [1]:

```
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python docker image: https://github.com/kaggle/docker-python
# For example, here's several helpful packages to load in

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('fivethirtyeight')
import networkx as nx
import warnings
warnings.filterwarnings('ignore')

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Enter) will list the files in the input directory

import os
print(os.listdir("../input"))

# Any results you write to the current directory are saved as output.
```

```
['kaggle-survey-2018', 'kaggle-survey-2017']
```

unfold_lessHide code

In [2]:

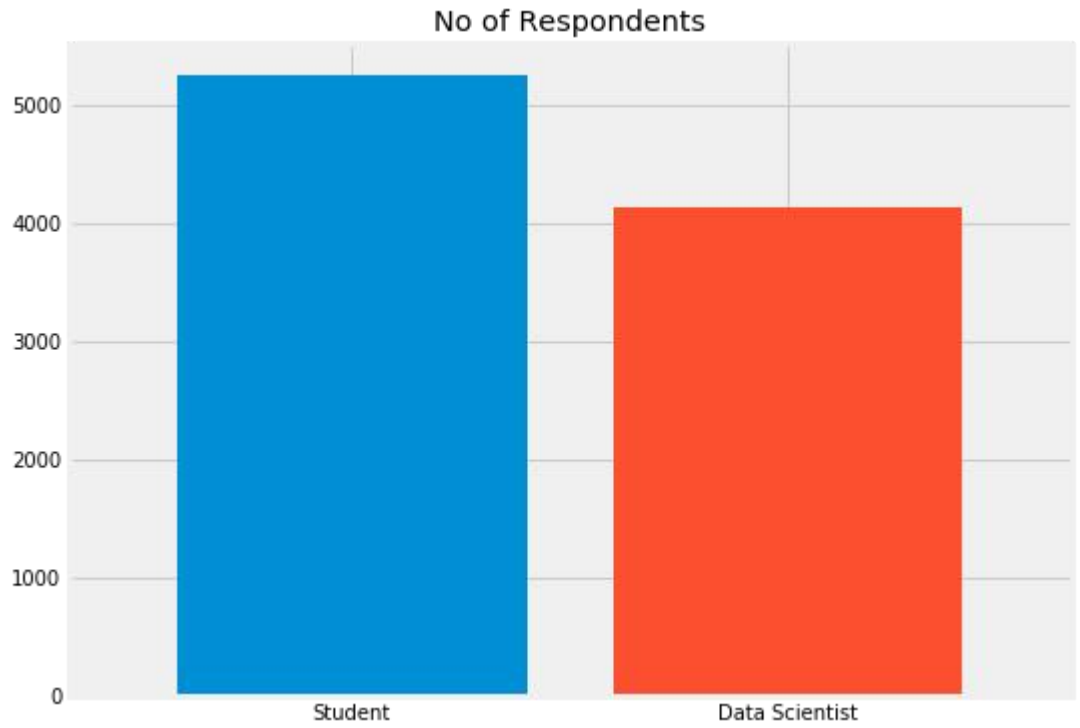
```
df=pd.read_csv('../input/kaggle-survey-2018/multipleChoiceResponses.csv')

viz=pd.read_csv('../input/kaggle-survey-2017/multipleChoiceResponses.csv',encoding='ISO-8859-1')

df.columns=df.iloc[0]

df=df.drop([0])
```

So before we start our questions, let us check how many students and Data Scientists have responded to the survey.

unfold_lessHide code

In [3]:

```
stu_ds=df[df['Select the title most similar to your current role (or most recent title if retired): - Selected Choice'].isin(df['Select the title most similar to your current role (or most recent title if retired): - Selected Choice'].value_counts()[:2].index)]

stu_ds['Select the title most similar to your current role (or most recent title if retired): - Selected Choice'].value_counts().plot.bar(width=0.8)

plt.gcf().set_size_inches(8,6)

plt.title('No of Respondents')

plt.xticks(rotation=0)

plt.show()
```



So there are pretty good number of responses from Data Scientists as compared to students. This will help us capture better analysis from a wide variety of respondents.

unfold_lessHide code

In [4]:

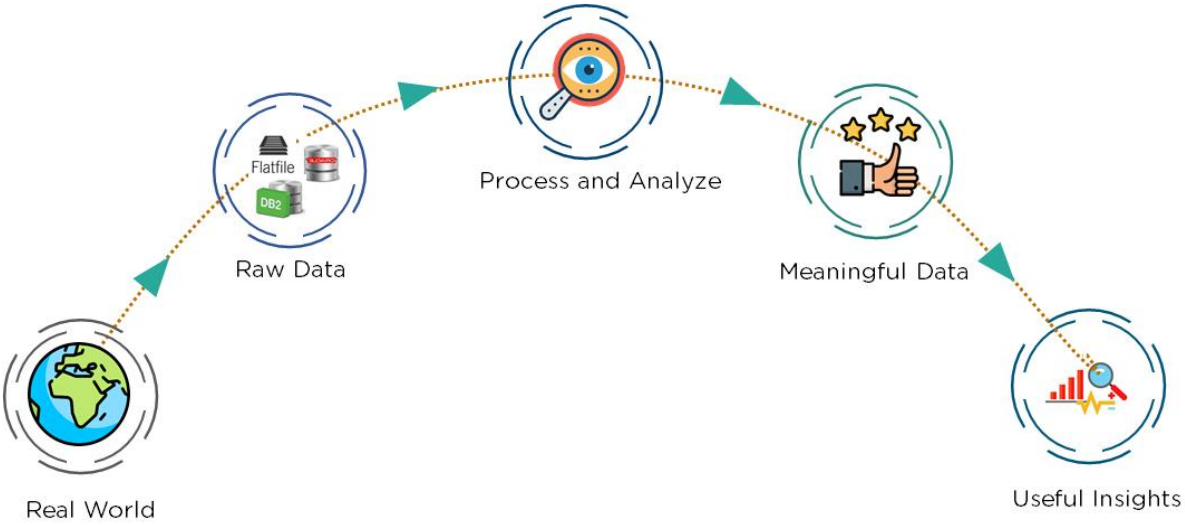
```
ds=stu_ds[stu_ds['Select the title most similar to your current role (or most recent title if retired): - Selected Choice']=='Data Scientist']

stu=stu_ds[stu_ds['Select the title most similar to your current role (or most recent title if retired): - Selected Choice']=='Student']
```

Q1. What is Data Science?

So there has been a lot of hype around Data Science, and everyone is kind of pushing themselves for becoming a Data Scientist. I too want to explore this field and tried researching about the same, but I still don't get a clear idea as some terminologies do sound like a jargon. So can you please explain what exactly Data Science is in simpler terms??

Answer:

In the simplest terms, Data Science is nothing but finding out meaningful insights and hidden patterns from a given trove of data. These insights and patterns prove to be very valuable for any business, as they can help in revenue generation, cost optimization and many other solutions. The diagram below will give you a better picture about the same.



And as the amount and variety of data is increasing, the responsibilities have increased and people are looking for experts and thus the demand for Data Scientists is also increasing. Thats the reason there's such a hype for Data Science. Just look at the interest trend for Data Science over the years.

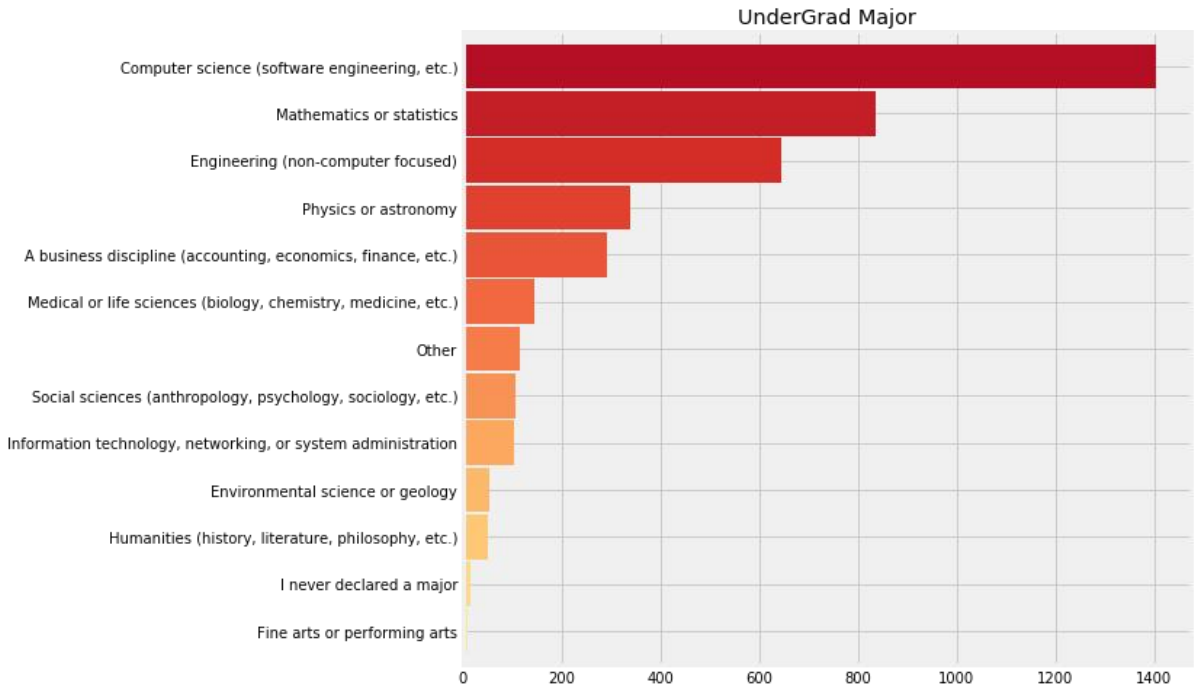
Q2. I am from a Non-Cs Background. Can I become a Data Scientist?

I belong to a Non-Computer Science background, and after hearing your previous answer I feel like only CS folks can become Data Scientists as they have an upper-hand in programming and using all the required tools. How can folks from Non-Cs background stand a chance with the CS folks?

Answer:

That's not the case at all. Data Science is a very diverse field, and it requires expertise and knowledge in a wide variety of domains like Statistics, Maths,etc. Just have a look at the below graph:

unfold_moreShow hidden code



You see, a majority of the Data Scientists belong to the Non-Cs background. The reason is that DS and ML can be applied to almost any field and its not just restricted to the CS folks. Also the classic job description many associate with a Data Scientist is A Statistician who can code!! So the CS folks might have a headstart in some tools and programming, but these can be learnt by anyone. Applying the knowledge for the correct business problems is what makes a great Data Scientist and not your college major.

Q3. Does holding an advanced Degree help you in Data Science?

Does holding an advanced degree like a Masters Degree or a PHD help you in this field as many employers don't even consider some resumes without such degree? If yes, what

recommendations would you give someone looking to make Data Science their career with little to no prior background?

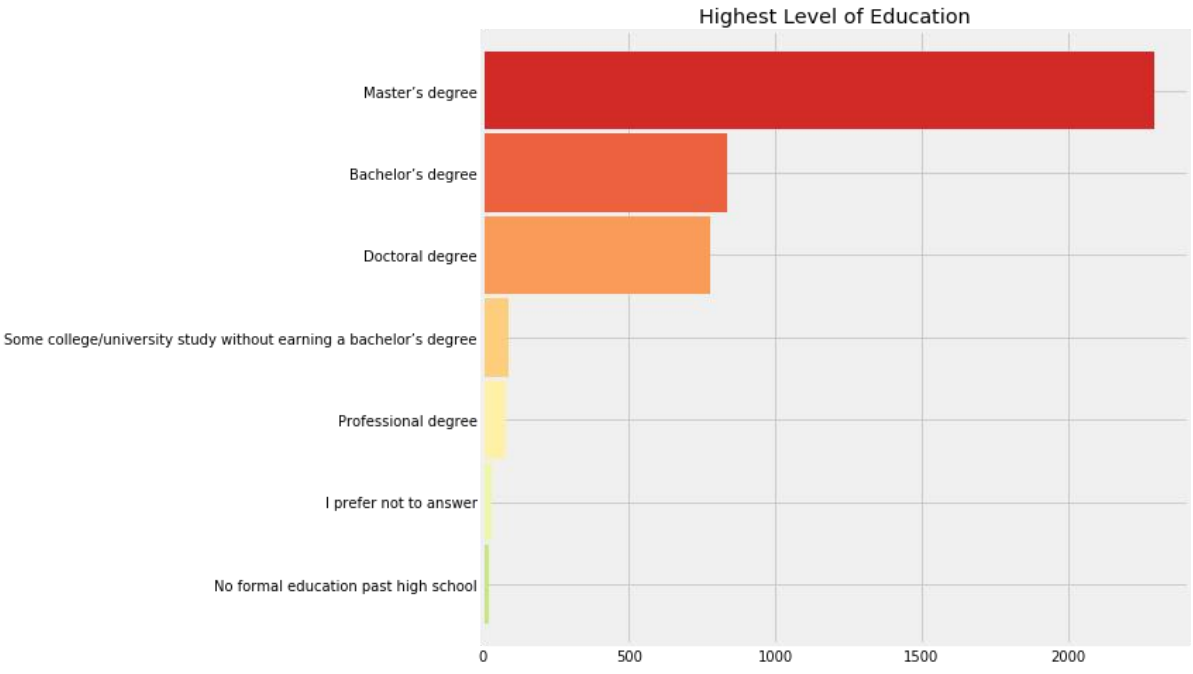
Answer:

This is a pretty interesting question!! For the first part of the question, I would say that the answer is yes in many cases. If you look, many of the Data Scientists hold a Masters Degree:

unfold_lessHide code

In [6]:

```
ds['What is the highest level of formal education that you have attained or plan to attain within the next 2 years?'].value_counts().plot.barh(width=0.95,color=sns.color_palette('RdYlGn',10))
plt.gcf().set_size_inches(8,8)
plt.gca().invert_yaxis()
plt.title('Highest Level of Education')
plt.show()
```



People do seek for an advanced degree as it is one of the best, if not the best, ways to showcase your expertise and knowledge. And as you already said, many recruiters do auto screen resumes i.e just discard them with no advanced degree. But there is absolutely nothing to worry about.

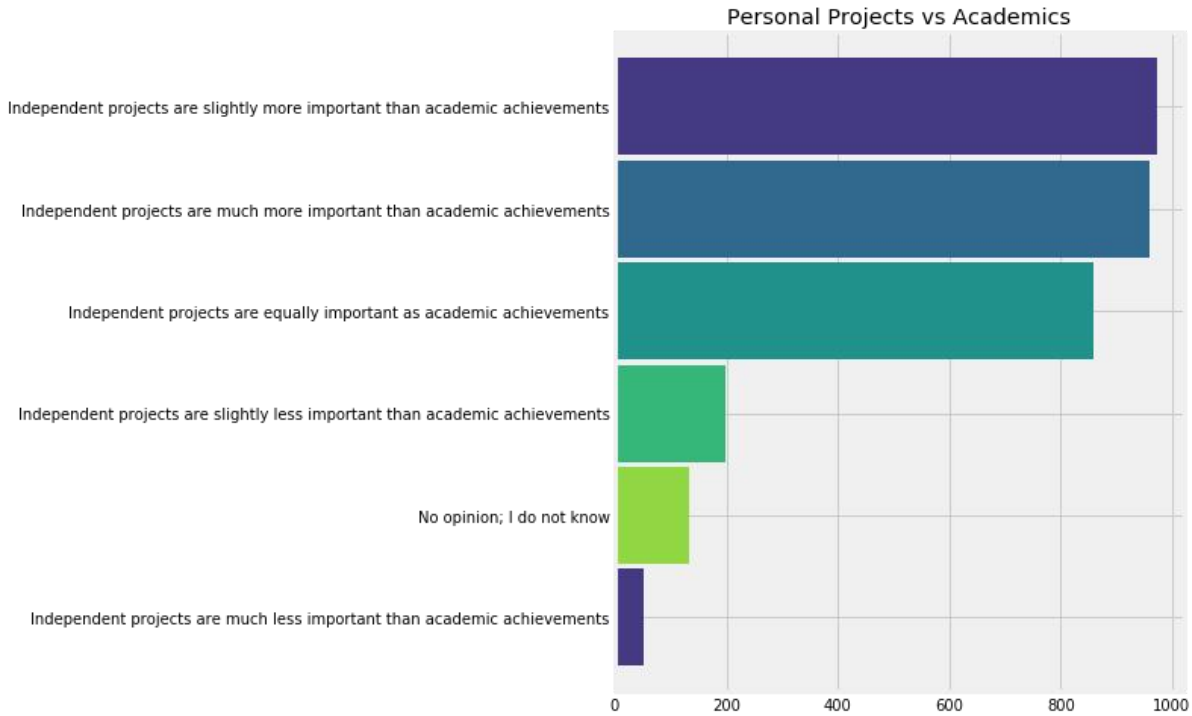
DJ Patil,former Chief Data Scientist of the United States Office of Science and Technology Policy once said: Data science is a team sport, and companies don't expect people to be unicorns. Technical experience is necessary but lots of companies we've talked with look more for other qualities (like curiousity, tenacity, open-mindedness, ability to pick up technical concepts, ability to work with others) than a specific set of qualifications. Doing data science in the real world is messy and uncertain, and if you can highlight your ability to overcome that kind of uncertainty and deliver, you'll go far. Anything that demonstrates that ability, from your research or personal projects would be great.

So I guess DJ Patil has already answered your second question!! Your research work or your personal projects also play a very pivotal role in your DS career, and I truly believe other Data Scientists also have a similar belief:

unfold_lessHide code

In [7]:

```
ds['Which better demonstrates expertise in data science: academic achievements or independent projects? - Your views:'].value_counts().plot.barh(width=0.95,color=sns.color_palette('viridis',5))
plt.gcf().set_size_inches(6,8)
plt.title('Personal Projects vs Academics')
plt.gca().invert_yaxis()
plt.show()
```



You see!! >90% of the Data Scientists believe that Personal Projects are important to showcase your skills and expertise as compared to your Academics. So its not really just degree or projects, but depends on the company you are applying for, the experise and experience they are looking for,etc. Hope that helps!!

Q4. Which Language should I learn first?

I have almost no experience in coding. Which language would you suggest for an aspiring Data Scientist like me to learn first?

Answer:

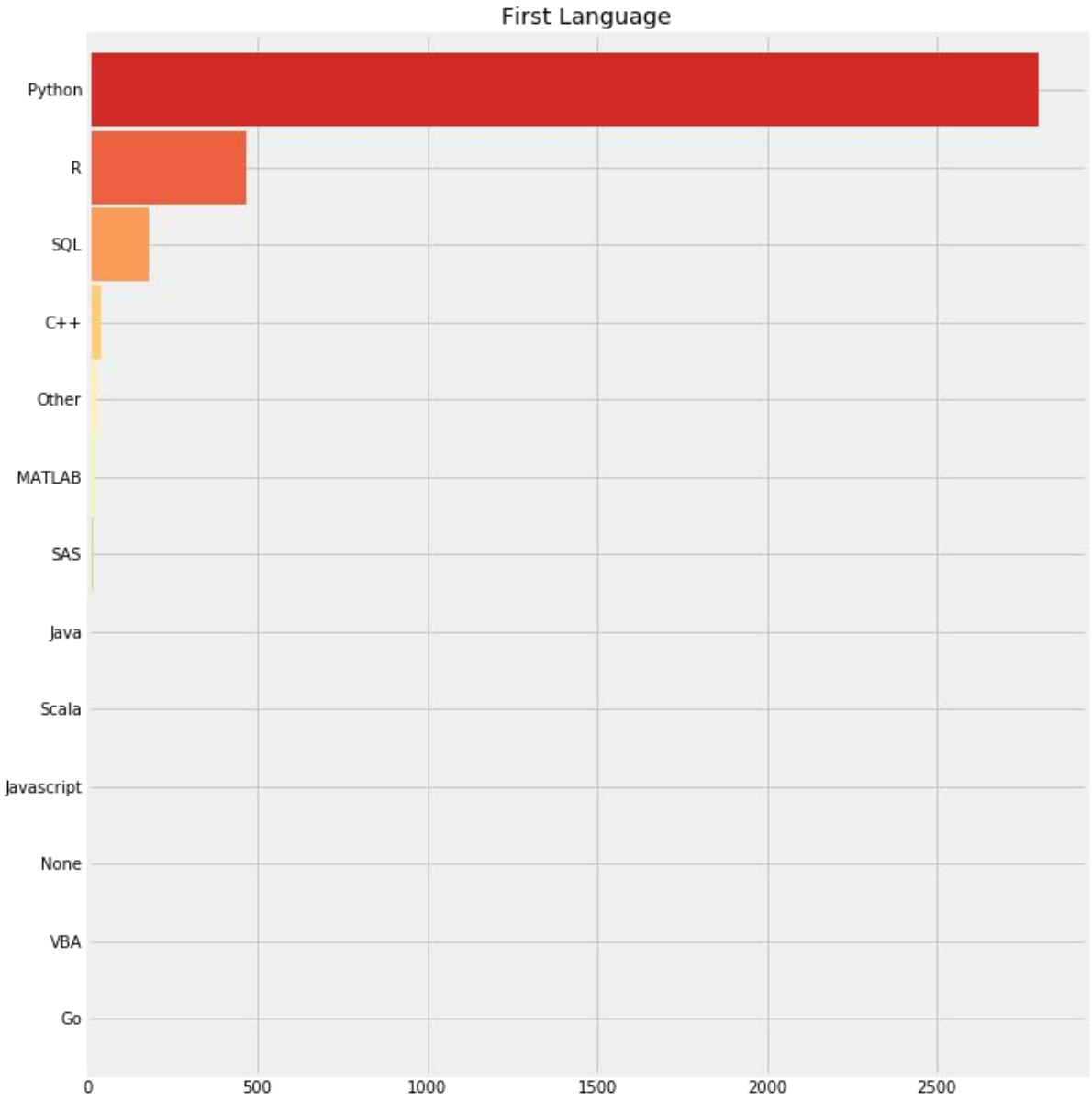
This questions is pretty simple and straightforward and I highly doubt anyone would disagree with my answer. And my answer for this question would be Python!!

unfold_lessHide code

In [8]:

```
ds['What programming language would you recommend an aspiring data scientist to learn first? - Selected Choice'].value_counts().plot.barh(width=0.95,color=sns.color_palette('RdYlGn',10))
```

```
plt.gcf().set_size_inches(10,12)
plt.gca().invert_yaxis()
plt.title('First Language')
plt.show()
plt.show()
```



Almost 90% of the people support my answer :p. Given its relatively shallow learning curve and increasingly robust data stack, I would say Python is a great tool to start. The

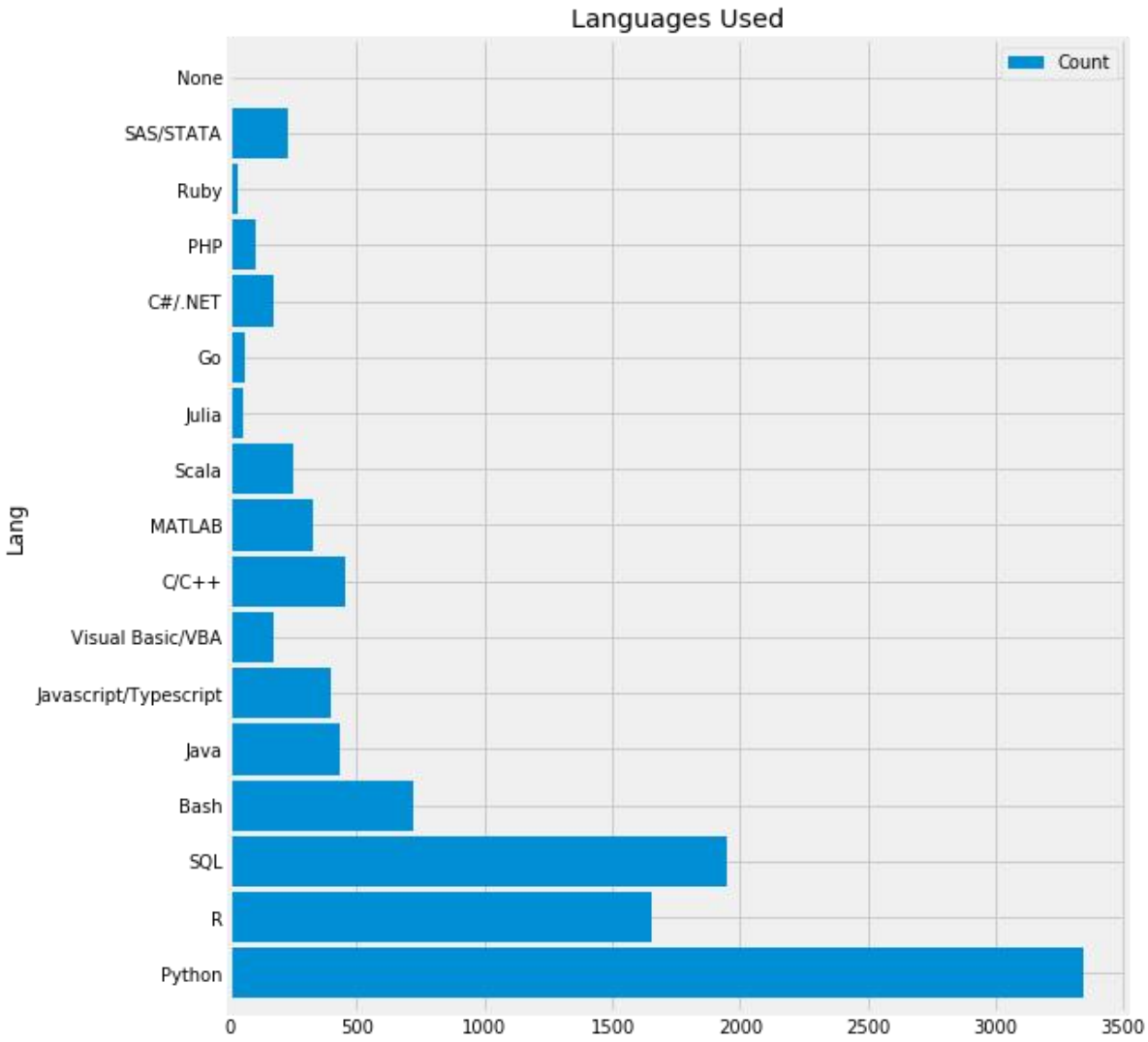
demand for Python has also gone up pretty well, and it did overtake R and other statistical tools in the current market.

However you should not restrict yourself to a single language or tool, as requirements change and you might also need to change your tools according to the new problems. So after learning Python, you can look into other important languages like SQL,R,etc. Have a look at which other languages/tools do the Data Scientists use on regular basis:

unfold_lessHide code

In [9]:

```
l1=[col for col in ds if col.startswith("What programming languages do you use on a regular basis? (Select all that apply)")]
col1=[]
col2=[]
l2=ds[l1[:-2]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
lang=pd.DataFrame({'Lang':col1,'Count':col2})
lang.set_index('Lang').plot.barh(width=0.9)
plt.gcf().set_size_inches(8,10)
plt.title('Languages Used')
plt.show()
```



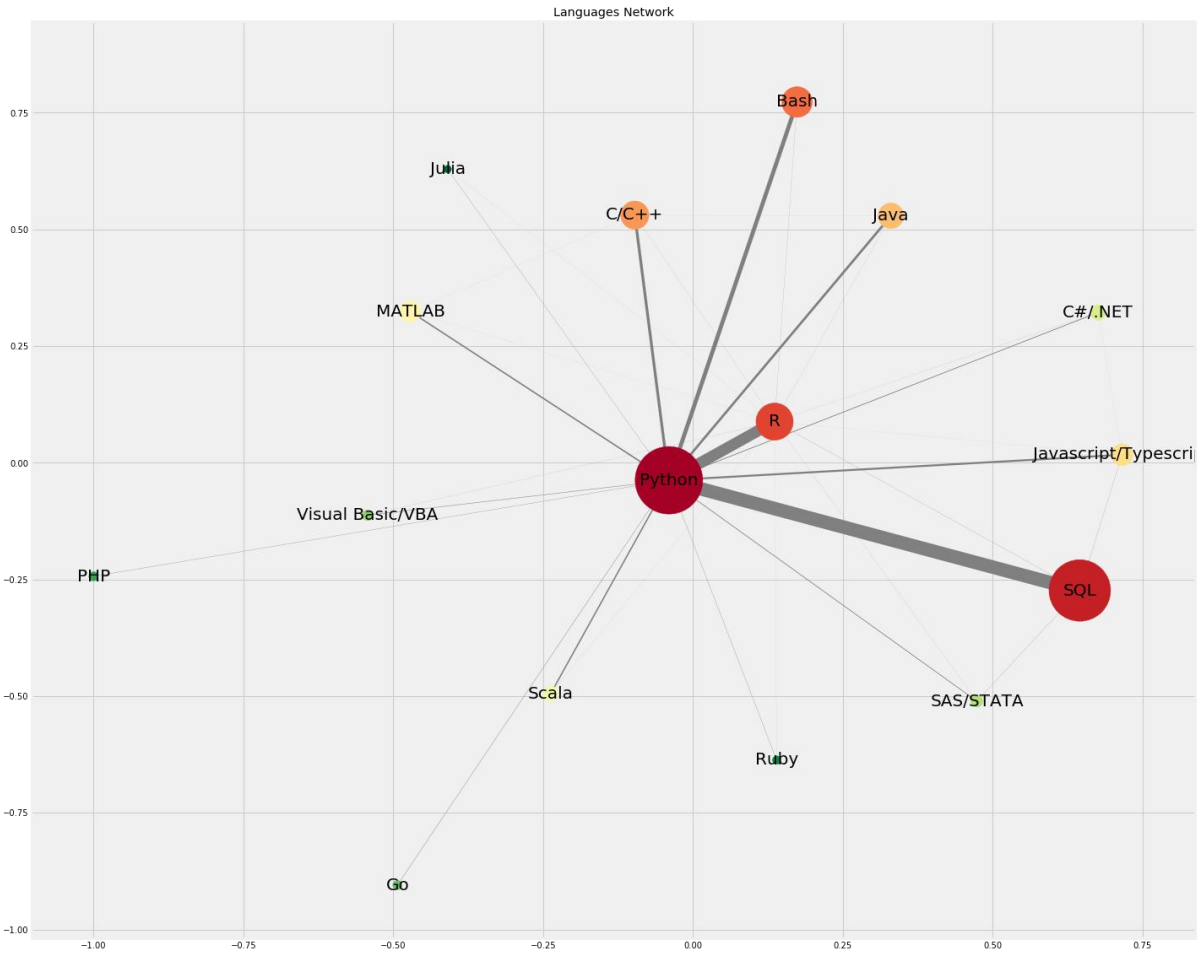
So these are the languages/tools we use in our daily work and it really depends on the task we are doing. For a even better idea, have a look at the below network chart.

unfold_lessHide code

In [10]:

```
lang=[col for col in ds if col.startswith('What programming languages do you use on a regular basis? (Select all that apply) - Selected Choice -')]

lang=lang[:-2]

df_lang=ds[lang]

df_lang.columns=[i[102:] for i in lang]

c = df_lang.stack().groupby(level=0).apply(tuple).value_counts()

out = [i + (j,) for i, j in c.items()]
```

```
out=[word for word in out if len(word)==3]

lang_net=pd.DataFrame(out)

lang_net.columns=['Lang 1','Lang 2','Count']

g = nx.from_pandas_edgelist(lang_net,source='Lang 1',target='Lang 2')

cmap = plt.cm.RdYlGn

colors = [n for n in range(len(g.nodes()))]

k = 0.35

pos=nx.spring_layout(g, k=k)

nx.draw_networkx(g,node_size=lang_net['Count'].values*20, cmap = cmap, node_color=colors,
edge_color='grey', font_size=20, width=lang_net['Count'].values*0.05)

plt.title('Languages Network')

plt.gcf().set_size_inches(22,20)
```



You see, there is a lot of bonding between Python,R and SQL, as they are the bread and butter for any Data Scientist. So the first language I would recommend to any aspiring Data Scientist is definetely Python. However they should also have a fair idea of the other major languages like SQL,R and others for becoming a better Data Scientist.

Q5. From where should I start learning from?

The Internet is now full of online courses that teach you data science. Some recommend to follow a course, while others suggest something else. There are lots of pathways which we can follow, so it is kind of overwhelming for us to figure out how to get started. So my question is what resources did you follow to teach yourself the requisite skills?

Answer:

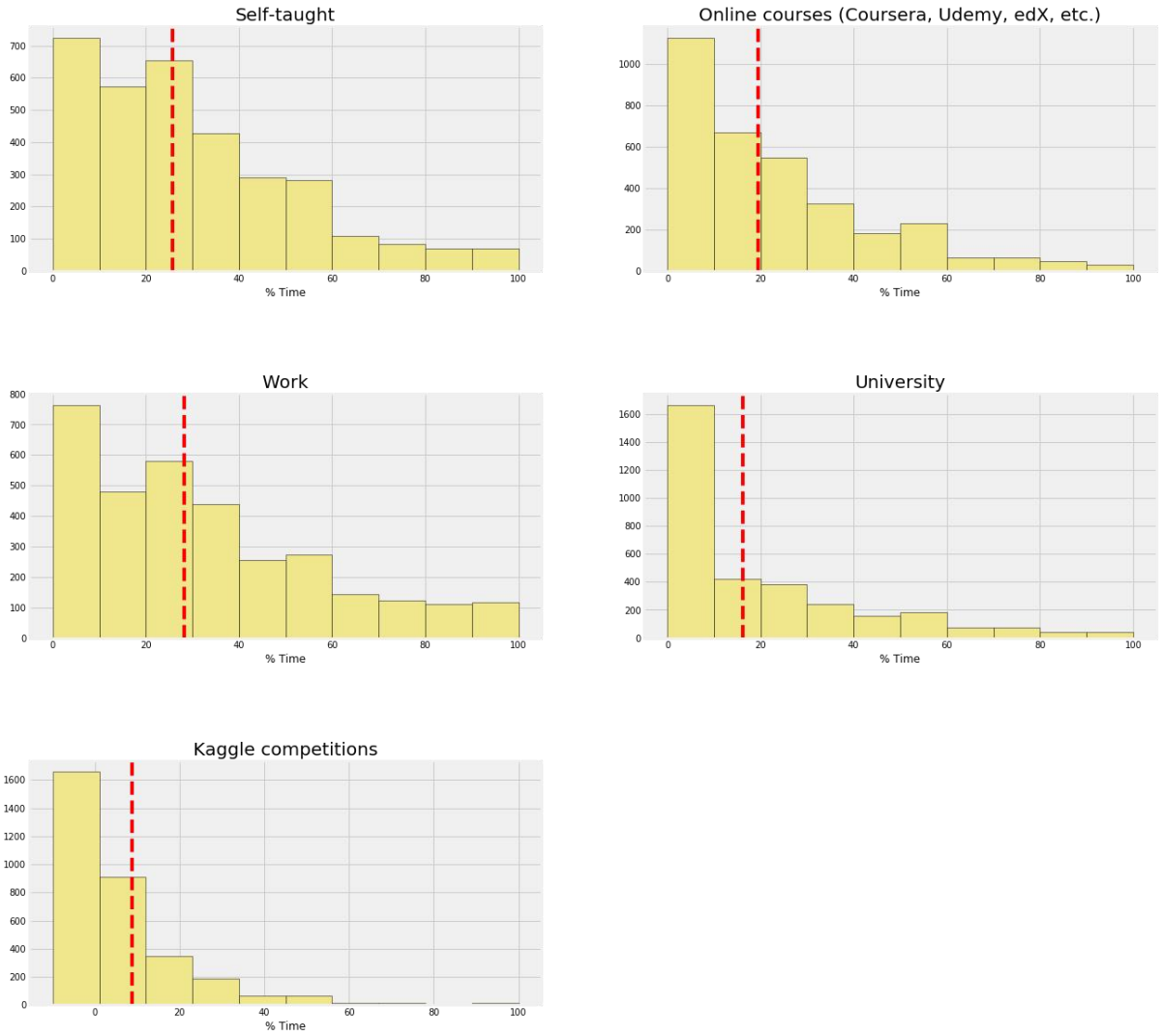
Great question!! For resources, I think it really depends a lot on your preferences. There are some great MOOC's available, and many people love them. However many prefer reading textbooks or just go through hundreds of research papers to acquaint themselves. For more applied practice, Kaggle is really a great arena to test your skills against the Top Class Data Scientists. However according to my personal opinion, your projects or your work help you learn and grasp concepts even better than any other source. Lets see how true it is!!

unfold_lessHide code

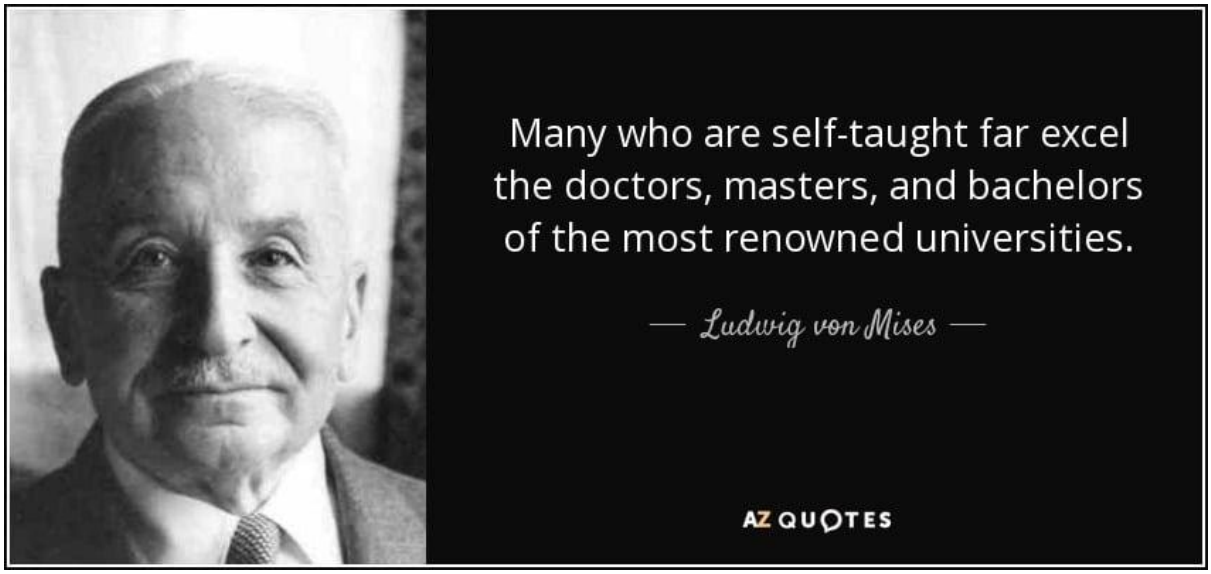
In [11]:

```
import itertools
import math
train=[col for col in ds if col.startswith('What percentage of your current machine learning/data science training falls under each category? ')]
train=train[:-2]
plt.figure(figsize=(20,20))
length=len(train)
for i,j in itertools.zip_longest(train,range(length)):
    plt.subplot(math.ceil((length/2)),2,j+1)
```

```
    plt.subplots_adjust(wspace=0.2,hspace=0.5)
    ds[i].astype('float').hist(bins=10,edgecolor='black',color='khaki')
    plt.axvline(ds[i].astype('float').mean(),linestyle='dashed',color='r')
    plt.title(i[130:],size=20)
    plt.xlabel('% Time')
plt.show()
```



You see!! Your work or your projects prove to be the real learning in any field. In real world, you won't get clean/tidy data and have to work with messy and humungous data. Dealing with such data proves to be real challenge, which in turn is a great learning experience.So your experience is the best way to learn Data SCience. Besides that Self-Teaching is another great way of learning.



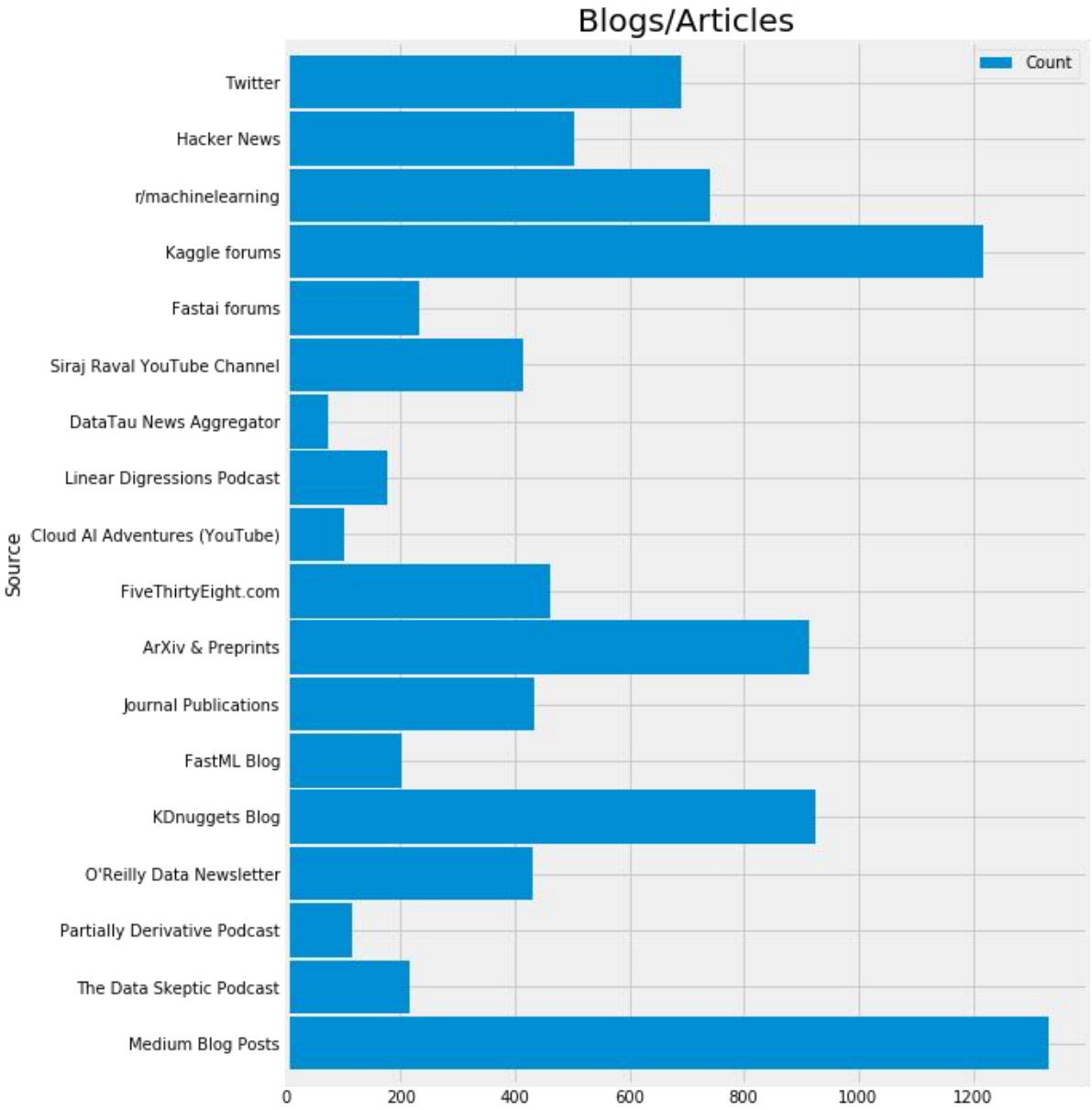
So what is Self-Teaching? Its nothing but just reading about every relevant thing you find, clearing your doubts on your own and implementing it.Personal Projects also fall under self-learning. Reading through articles, blogs ,technical papers are some of the best ways to educate yourself. Some of the best resources/blogs/articles you can look out for:

unfold_lessHide code

In [12]:

```
l1=[col for col in ds if col.startswith('Who/what are your favorite media sources that report on data science topics? (Select all that apply) - Selected Choice -')]
col1=[]
col2=[]
l2=ds[l1[:-4]]
l2
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
activity=pd.DataFrame({'Source':col1,'Count':col2})
activity.set_index('Source').plot.barh(width=0.95)
plt.gcf().set_size_inches(8,12)
plt.gca().invert_yaxis()
plt.title('Blogs/Articles',size=20)
```

plt.show()

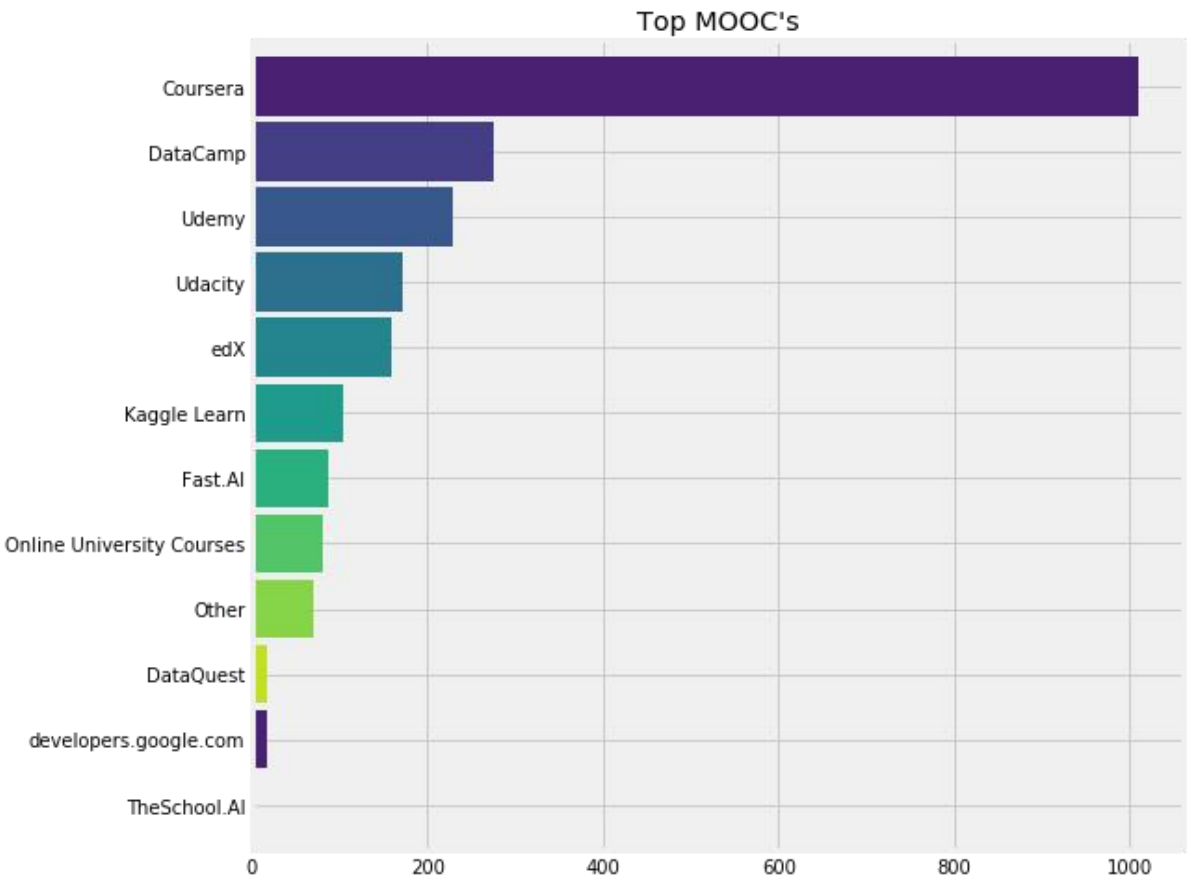


However if you are not the kind of person who likes exploring on their own and need a guide or a structured way of learning, you can switch to online courses. MOOC's have become a buzzword in all domains, and you will find hundreds of courses teaching you fundamentals as well as advanced concepts of Data Science. So going through some basic courses and then implementing stuff can also be a very good direction. Some of the best MOOC's:

unfold_lessHide code

In [13]:

```
ds['On which online platform have you spent the most amount of time? - Selected Choice'].value_counts().plot.barh(width=0.9,color=sns.color_palette('viridis',10))

plt.gcf().set_size_inches(8,8)

plt.gca().invert_yaxis()

plt.title("Top MOOC's")

plt.show()
```



So all in all if you are serious about making a move in this field, self-teach yourself(reading through basic conceptual blogs/articles) for 2-3 hours a day after work/studies for at least a few months, or start with some online cources. Once you feel like you are comfortable with the concepts, move onto more applied work, try

implementing your own projects and reflect them in your resume or you can even take up some Kaggle Competitions.I hope that helps!!

Q6. Being a Data Scientist, which all industries can I work in?

As Data Science is becoming mainstream, are all the indutries embracing it, or are there only a few industries accepting it. Also if I want to work in a specific industry, what more should I be prepared with?

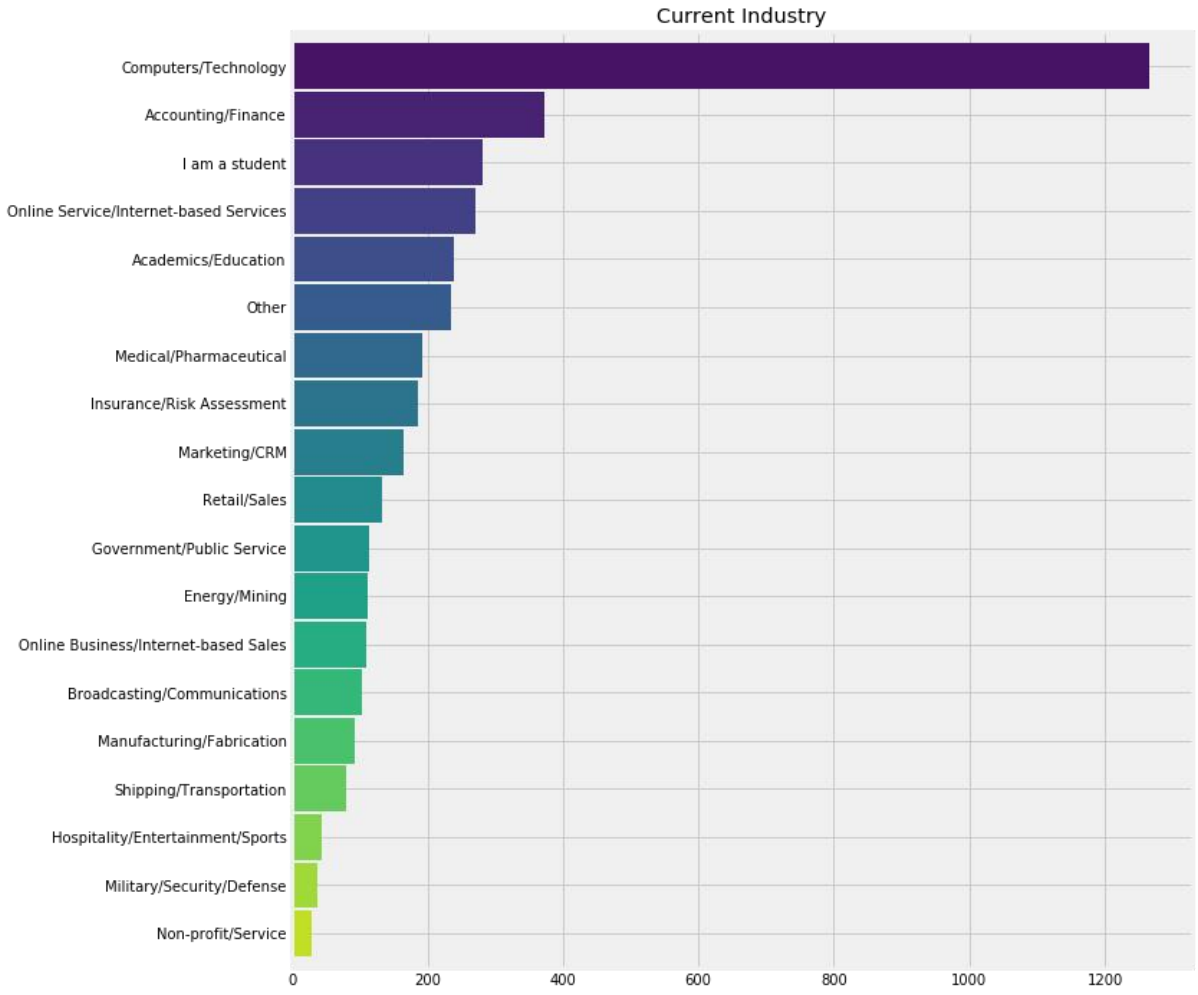
Answer:

As I had already mentioned, DS and ML are a very broad field, and they can be applied to almost all industries. It still hasn't become popular in all the industries, but the rate of acceptance is increasing.Being a Data Scientist, you can work in:

unfold_lessHide code

In [14]:

```
ds['In what industry is your current employer/contract (or your most recent employer if retired)? - Selected Choice'].value_counts().plot.barh(width=0.95,color=sns.color_palette('viridis',20))
plt.gcf().set_size_inches(10,12)
plt.gca().invert_yaxis()
plt.title('Current Industry')
plt.show()
```



ML is heavily used in the Computers/Technology domain, with all the research work in AI/ML in this domain, the requirements for DS/ML experts will increase exponentially. Demand for Data Scientists is also ought to increase in other industries.

Now coming to your 2nd question, make sure that you have at least basic domain knowledge specific to the company/industry.A lot of DS jobs these days involve the 'product' side. For example, if you are aiming for a financial/marketing industry, have a fair bit of idea on some financial jargons like forecasting,churn rates,securities,etc. If you are aiming for a CS company, you may be questioned about basic data structures and alogorithm problems. So having a basic knowledge in that domain can prove to be fruitful in any interview.

Q7. Can you give a brief idea about your daily activities/tasks performed?

I am very curious to know what all tasks do you guys perform on daily basis. Can you throw some light on the various activities you perform daily and how important they are?

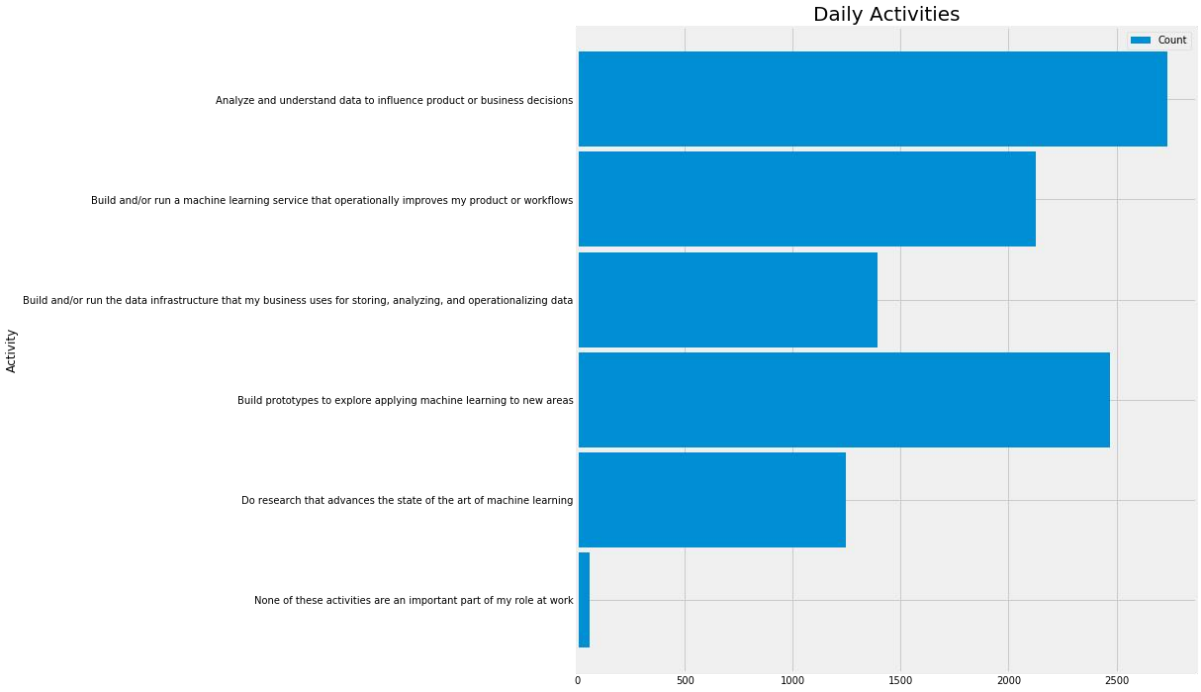
Answer:

Data science has profound impacts on a business, and thus analysing data and finding out the possible impacts of any action on the business, refining the existing models,etc are some of the activities we perform daily. Other important activities take make up the day for a Data Scientist are:

unfold_lessHide code

In [15]:

```
l1=[col for col in ds if col.startswith("Select any activities that make up an important part of your role at work: (Select all that apply) ")]
col1=[]
col2=[]
l2=ds[l1[:-2]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
activity=pd.DataFrame({'Activity':col1,'Count':col2})
activity.set_index('Activity').plot.barh(width=0.95)
plt.gcf().set_size_inches(10,12)
plt.gca().invert_yaxis()
plt.title('Daily Activities',size=20)
plt.show()
```



Analysing and understanding data for business process is the most important activity performed daily by any Data Scientist.An example that glorifies the importance of data analysis can be drawn from the movie Moneyball. The movie showed how old ways of evaluating performance in baseball were outperformed by the application of data science. One baseball team used data science techniques to overcome its financial disadvantage. It achieved this by using analytics to identify high-performing players who other teams had overlooked using traditional methods, and therefore acquired their services at a relatively low cost. The result was that the team regularly beat higher-spending competitors in their league.

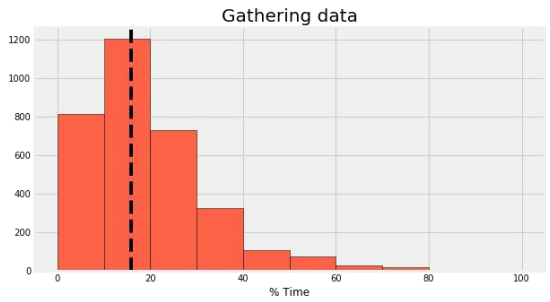
Another important job is prototyping ML models for application in newer domains/areas.Human decision making can be inaccurate.With the ever-expanding universe of data,we can use DS and especially ML,we can build prototypes that can be later be developed into products, to solve highly complex data-rich problems that can overwhelm even the smartest person. Refinement is another common activities for Data Scientists. Most data scientists work in the production part of their business and have established models for refining processes and products according to the data their organization collects. Common examples can be marketing segmentation,banks adjusting their financial risk models,etc.

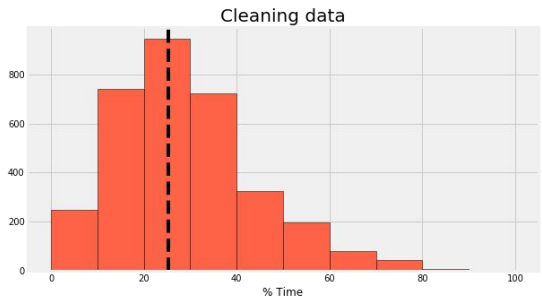
Apart from these tasks, we spend a lot of time in data management like data collection, cleaning and many such challenging tasks:

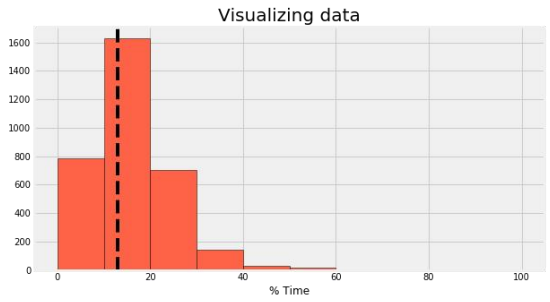
unfold_lessHide code

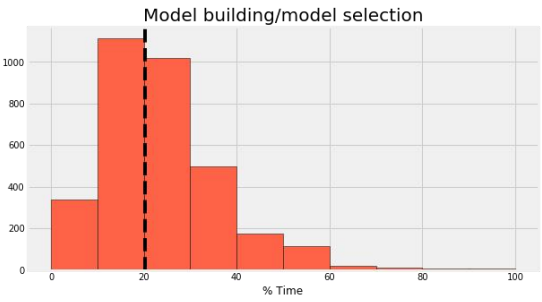
In [16]:

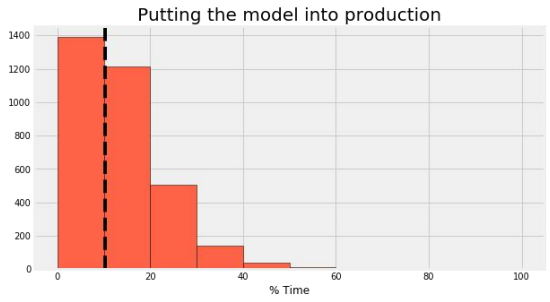
```
time_spent=[col for col in ds if col.startswith("During a typical data science project at work or school, approximately what proportion of your time is devoted to the following? ")]
time_spent=time_spent[:-1]
plt.figure(figsize=(20,20))
length=len(time_spent)
for i,j in itertools.zip_longest(time_spent,range(length)):
    plt.subplot((length/2),2,j+1)
    plt.subplots_adjust(wspace=0.2,hspace=0.5)
    ds[i].astype('float').hist(bins=10,edgecolor='black',color='tomato')
    plt.axvline(ds[i].astype('float').mean(),linestyle='dashed',color='black')
    plt.title(i[161:],size=20)
    plt.xlabel('% Time')
plt.show()
```

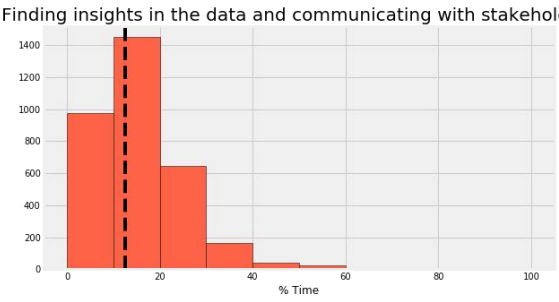












Gathering Data: Undoubtedly the one of the most time consuming part in the entire pipeline. Getting the data can be painstaking and it really depends from where do we fetch our data. If it comes from a publicly hosted site like Kaggle, it is very easy and no time is required. However its not the case in real world. In real world cases, we need to select the right data in the right format and build a secure way of transferring the data flow to our ML models or application.

Cleaning Data: The most time consuming part, and I am sure no one will disagree on this!! Transforming the data into correct format for the application, detecting and correcting corrupt or inaccurate data, etc all come under data cleaning. The main challenge in this is correction of values to remove duplicates and invalid entries, as deletion of data can lead to information loss. Hence critical decisions and thinking is required, which makes it a time consuming process.

Visualizing Data: It is probably the least time consuming process but a very important one, and it reduces even further if we use Enterprise Tools like Tableau,Qlik,Tibco,etc, which helps in building graphs and dashboards with simple drag and drop features.

Model Building: It is where the data scientists build decide a suitable algorithm ,build predictive models, tune these models,etc.

Putting Model into Production: Simply it means encapsulating the ML model into an application or hosting it somewhere so that others can use the model with their own data. Now a days it has become very easy to build API's that directly query the ML model and return results based on the user input data. The process has become easier due to the great integration of ML models and cloud deployment, which has infact reduced the time required for production deployment.

Finding Insights and Communicating with Stakeholders: Finding insights is finding hidden patterns and facts in the trove in data and effectively communicating it to the clients with minimum cognitive load, such that it helps improving business decisions and has a positive impact on the business processes. Effective communication and simple but effective visualizations play a very important role in this phase.

Q8. How important are Data Visualization skills for a Data Scientist?

Are Data Visualization skills an important one for a Data Scientist? What tools/libraries do you use for data visualisation?

Answer

Data Visualization is one of the most important task but is considered as one of the most undervalued task in the pipeline. Before diving into building models, every Data Scientist performs EDA(Exploratory Data Analytics),which is the practice of describing the data by means of statistical and visualization techniques to bring important aspects of that data into focus for further analysis. This involves looking at your data set from many angles, describing it, and summarizing it without making any assumptions about its contents. Some of the key benefits of visualizations are:

- It helps absorb information quickly
- You can share your insights with everyone
- Easily spot outliers,etc

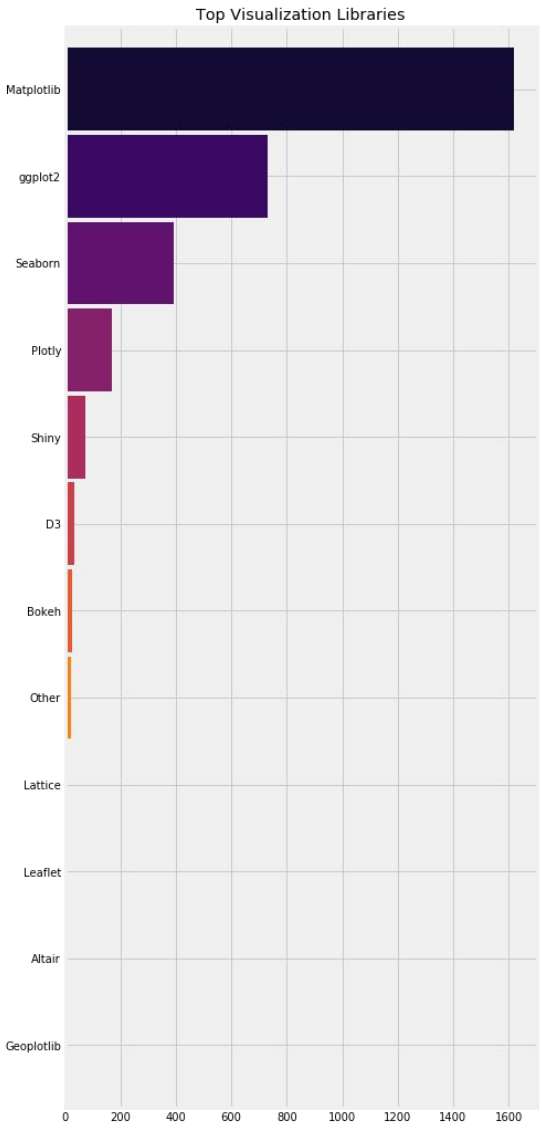
unfold_lessHide code

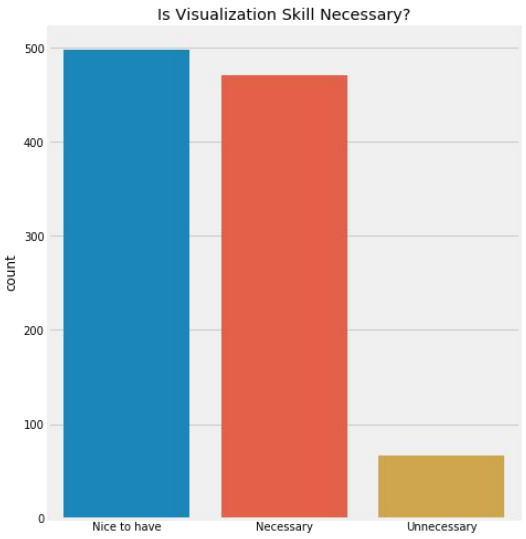
In [17]:

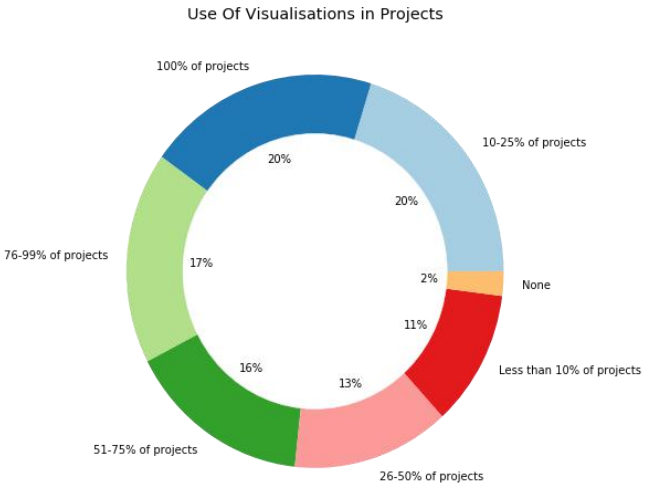
```
import matplotlib.gridspec as gridspec

scientist=viz[viz['DataScienceIdentitySelect']=='Yes']

fig = plt.figure(figsize=(15,18))

gridspec.GridSpec(2,2)


plt.subplot2grid((2,2), (0,0), colspan=1,rowspan=2)

ds['Of the choices that you selected in the previous question, which specific data visualization library or tool have you used the most? - Selected Choice'].value_counts().plot.barh(width=0.95,color=sns.color_palette('inferno',10))

plt.gca().invert_yaxis()

plt.title('Top Visualization Libraries')

plt.subplot2grid((2,2), (0,1))

sns.countplot(scientist['JobSkillImportanceVisualizations'])

plt.title('Is Visualization Skill Necessary?')

plt.xlabel('')


plt.subplot2grid((2,2), (1,1), colspan=1,rowspan=2)

scientist['WorkDataVisualizations'].value_counts().plot.pie(autopct='%2.0f%%',colors=sns.color_palette('Paired',10))

plt.title('Use Of Visualisations in Projects')

my_circle=plt.Circle( (0,0), 0.7, color='white')

p=plt.gcf()

p.gca().add_artist(my_circle)

plt.ylabel('')
```

Out[17]:

Text(0,0.5,'')







Visualisations are a very integral part of Data Science Projects, and being a Data Scientist you should have a good understanding of the same. Coming to the libraries, since Python and R are the leading tools today, you must learn Matplotlib and Seaborn in Python and GGplot2 in R. Not all libraries support all the various graphs, you might thus need to use other libraries/tools for very specific use cases.

Q9. Do you think in depth knowledge of Machine Learning Algorithms is necessary?

I have seen people using machine learning models without any in-depth knowledge or working of the algorithm, i.e they consider ML models as 'Black-Boxes'. Does this really work in your work/projects? Do I need to know the proper working of the algorithms? Also which all ML/DL libraries do you use in your work?

Answer:

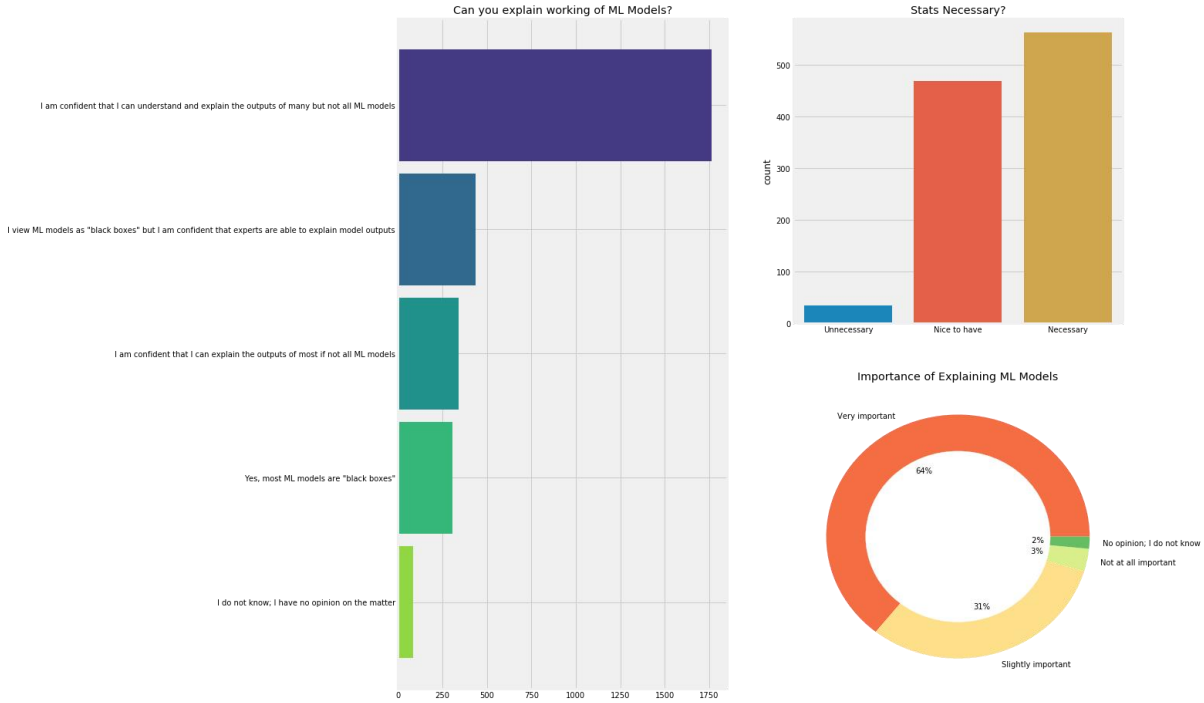
To a certain extent I think it is really beneficial to get a grasp on what's going on under the hood. You probably don't need to know the exact code implementation, but understanding how the algorithms work will be super useful.Other Data Scientists have a similar opinion:

unfold_lessHide code

In [18]:

```
fig = plt.figure(figsize=(15,16))

gridspec.GridSpec(2,2)


plt.subplot2grid((2,2), (0,0), colspan=1,rowspan=2)

ds['Do you consider ML models to be "black boxes" with outputs that are difficult or impossible to explain?'].value_counts().plot.barh(width=0.9,color=sns.color_palette('viridis',5))

plt.title('Can you explain working of ML Models?')

plt.gca().invert_yaxis()

plt.subplot2grid((2,2), (0,1))

sns.countplot(scientist['JobSkillImportanceStats'])

plt.title('Stats Necessary?')

plt.xlabel('')

plt.subplot2grid((2,2), (1,1), colspan=1,rowspan=2)

ds['How do you perceive the importance of the following topics? - Being able to explain ML model outputs and/or
predictions'].value_counts().plot.pie(autopct='%2.0f%%',colors=sns.color_palette('RdYlGn',4))

plt.title('Importance of Explaining ML Models')

my_circle=plt.Circle( (0,0), 0.7, color='white')
```

```
p=plt.gcf()
p.gca().add_artist(my_circle)
plt.ylabel('')
plt.show()
```



You see!! majority of the Data Scientists say that they can understand as well explain the outputs of the ML models. Not all ML models can be used for all types of data. Understanding the data and its statistical analysis is very important before applying any model. Thats the reason why the knowledge of stats is also necessary for ML modles. All in all, ML models should not be considered as black-boxes and you should have a good idea about the underlying mechanism.

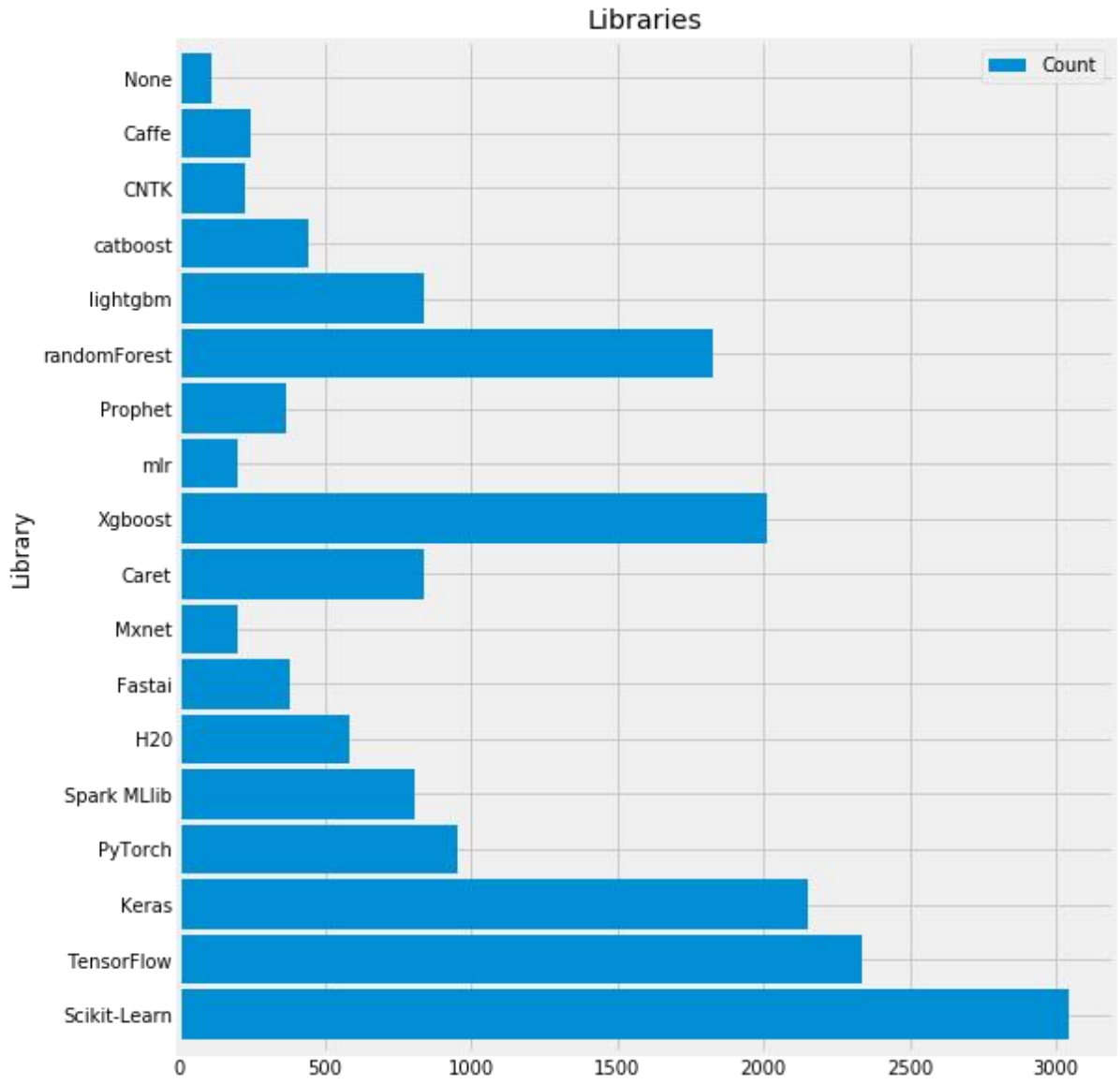
Now coming to the commonly used tools/libraries/frameworks:

unfold_lessHide code

In [19]:

```
l1=[col for col in ds if col.startswith("What machine learning frameworks have you used in the past 5 years? ")]
```

```
col1=[]
col2=[]
l2=ds[l1[:-2]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
lib=pd.DataFrame({'Library':col1,'Count':col2})
lib.set_index('Library').plot.barh(width=0.9)
plt.gcf().set_size_inches(8,10)
plt.title('Libraries')
plt.xlabel('')
plt.show()
```



Sklearn is one of the most sought after libraries in Python, as it contains all the traditional as well as many advanced ML algorithms built into a single API. For Deep Learning purposes, we have packages like Tensorflow,Keras,etc. Tensorflow is built by Google and Keras runs on top of TensorFlow. But the library shouldn't be a concern.In any case, we should just pick any library and focus more on concepts rather than just implementing them in one of the packages.

Tools change, new tools are being developed, and who knows what package will be the "best" one in a few years. Thus, focussing on concepts and being a bit flexible in terms of packages may not be a bad thing to do. Often, a single package is also not enough to do anything you want to do. So don't restrict to packages, go for concept clearance.

Q10. What is Bias in Machine Learning?

Answer:

Before we understand what is bias, lets imagine that we have developed some ML models that do the following:

- **Selecting students based on their caste/ethnicity.**
- **Banks providing loans based on color, or gender and not on financial score.**
- **Rejecting applications solely based on the sex, etc.**

What do you think are these models doing? Ain't they biased towards a particular group? So bias in ML models means that the model has been trained with data that is biased/influenced. Thus bias and fairness are two contradictory terms in Machine Learning. Bias can occur either due to data collection techniques, errors by data analysts, or the data itself maybe biased. ML models are trained on data collected by us, and thus it may contain inherent biases. Removing bias from data to make fair decisions is a key ML problem, and I strongly believe that others also agree to this point.

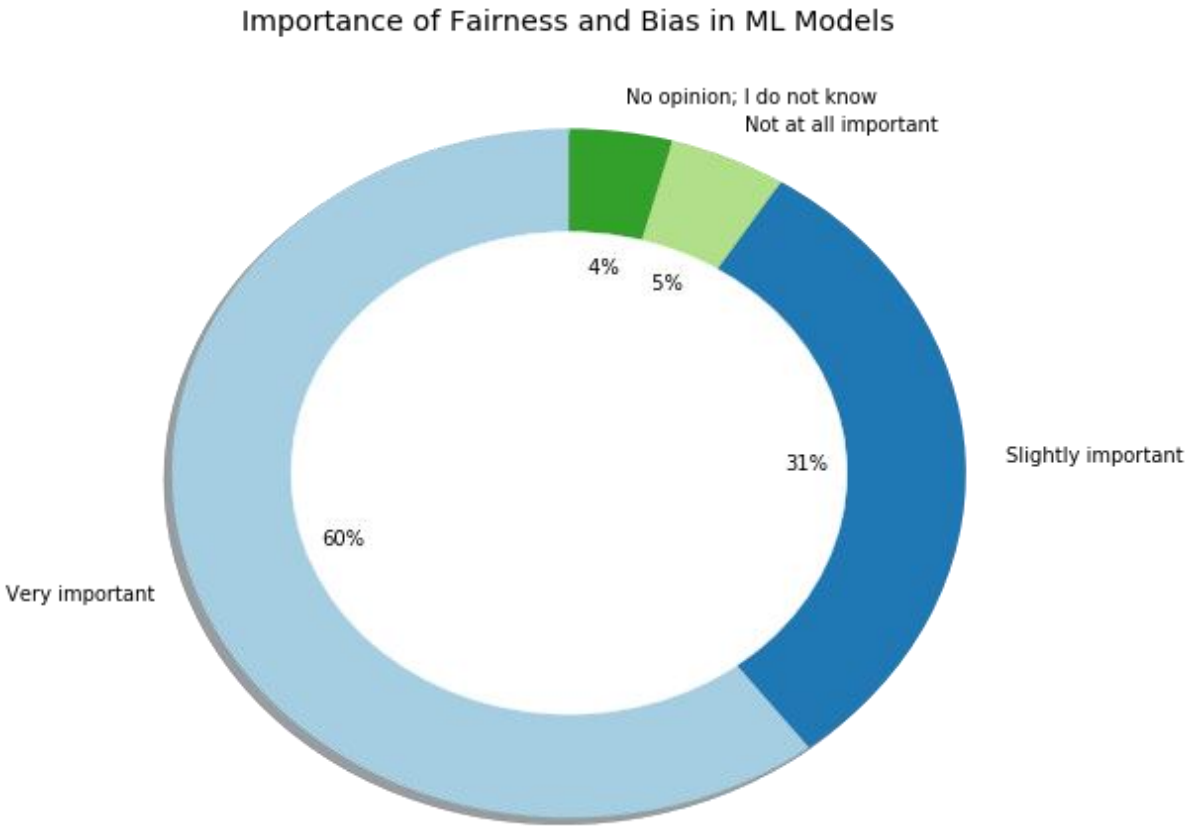
unfold_lessHide code

In [20]:

```
ds['How do you perceive the importance of the following topics? - Fairness and bias in ML algorithms:'].value_counts().plot.pie(autopct='%2.0f%%',shadow=True,startangle=90,colors=sns.color_palette('Paired',6))

plt.title('Importance of Fairness and Bias in ML Models')

my_circle=plt.Circle( (0,0), 0.7, color='white')

plt.gcf().set_size_inches(8,8)

plt.gca().add_artist(my_circle)

plt.ylabel('')
```

Out[20]:

Text(0,0.5,'')



However not many spend enough time uncovering the unwanted bias in the dataset.

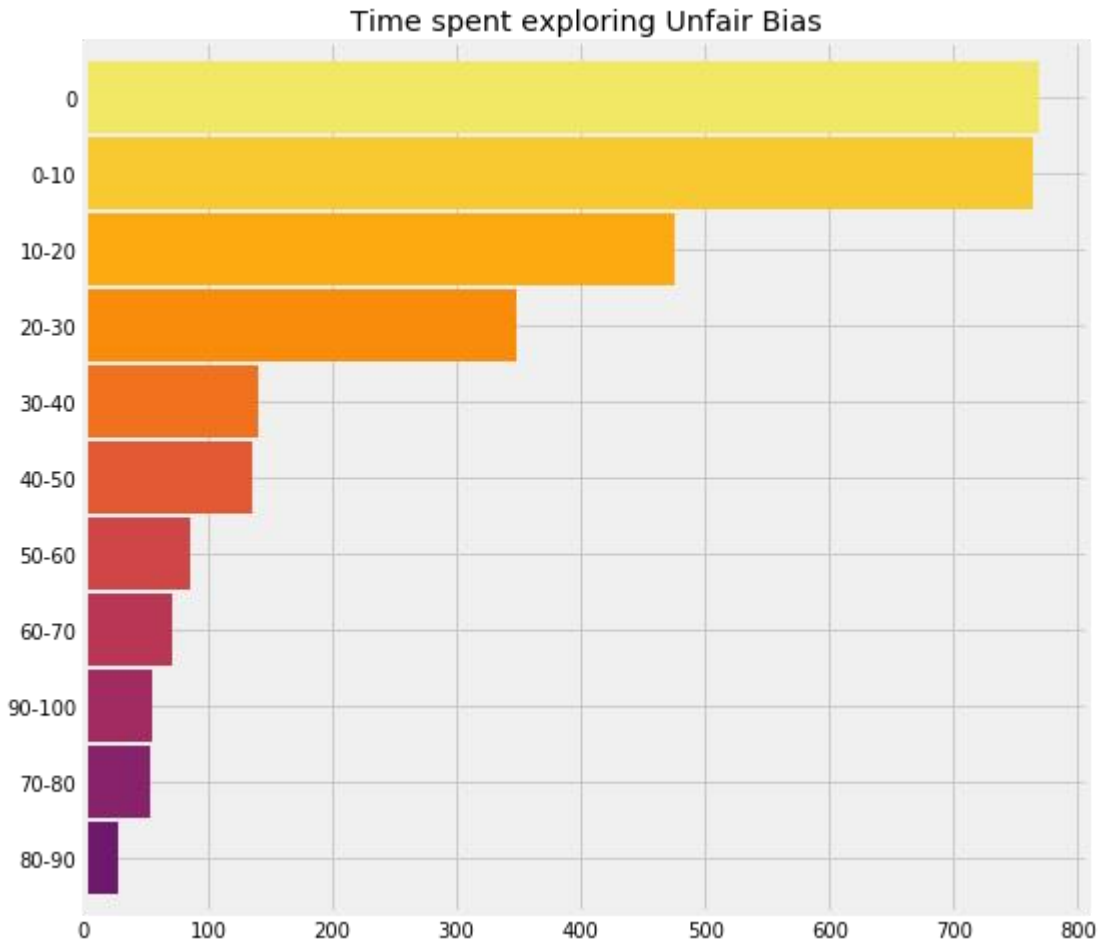
unfold_lessHide code

In [21]:

```
ds['Approximately what percent of your data projects involved exploring unfair bias in the dataset and/or algorithm?'].value_counts().plot.barh(width=0.95,color=sns.color_palette('inferno_r',15))

plt.gca().invert_yaxis()

plt.gcf().set_size_inches(8,8)

plt.title('Time spent exploring Unfair Bias')
```

Out[21]:

Text(0.5,1,'Time spent exploring Unfair Bias')



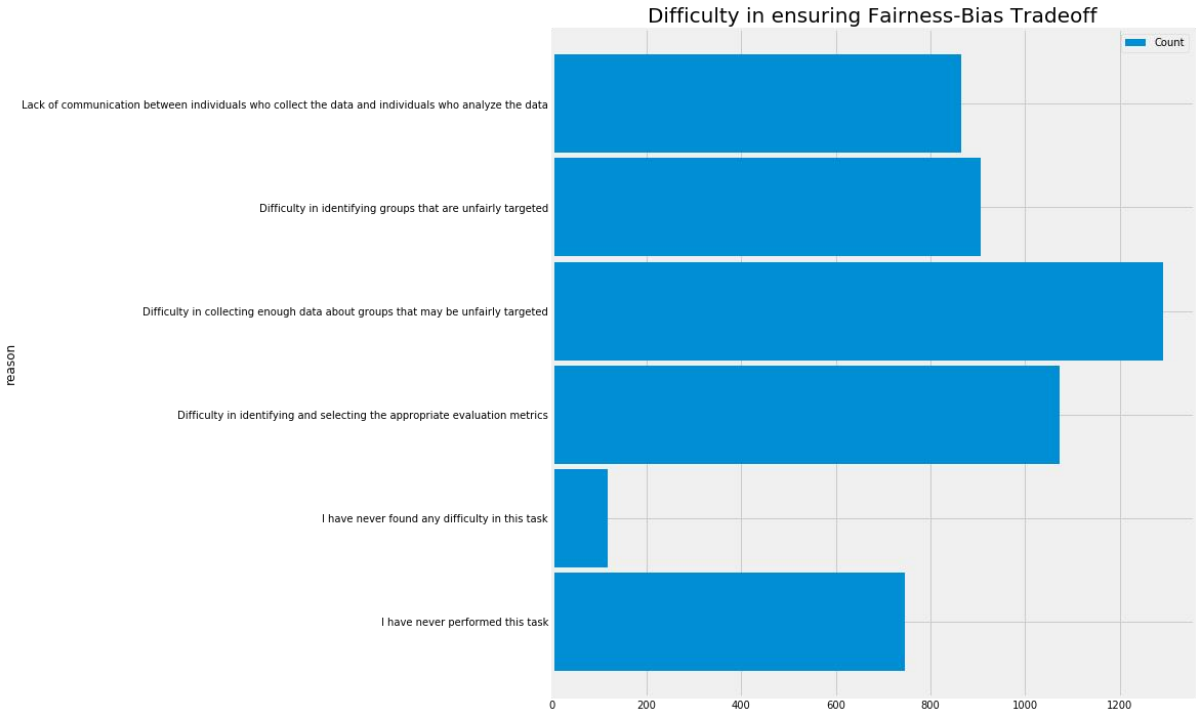
Some of the probable reasons that stops people from exploring the unwanted bias are:

unfold_lessHide code

In [22]:

```
l1=[col for col in ds if col.startswith("What do you find most difficult about ensuring that your algorithms are fair and unbiased?")]
col1=[]
col2=[]
l2=ds[l1[:]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
bias=pd.DataFrame({'reason':col1,'Count':col2})
```

```
bias.set_index('reason').plot.barh(width=0.95)
plt.gcf().set_size_inches(10,12)
plt.gca().invert_yaxis()
plt.title('Difficulty in ensuring Fairness-Bias Tradeoff',size=20)
plt.show()
```



Major reason for this problem is unbalanced data. Many a times you will find that data for a group is too large or too small as compared to the other groups in the dataset. What the model will do in this case is that it will be able to predict the class with larger data, and give wrong predictions for the class with less data. Unbalanced data can thus be used to unfairly target some groups, thereby inducing bias in the model.

Other reasons include lack of communication between the teams that analyse the data and the ones who use it for modeling. Biased sources of data or counterfiet sources of data can also be difficult to judge, and thus bias is generated implicitly in the models.

Q11. How do you explain the output of your model?

Do you conclude that your model is effective solely by looking at its accuracy, or are there any other metrics you check for?

Answer:

Always remember - ACCURACY CAN BE MISLEADING!! Evaluating and explaining your machine learning algorithm is an essential part of any project. Your model may give you a good accuracy while testing, but it may fail when you finally deploy it into your business processes. Most of the times we use accuracy to measure the performance of our model, however it is not enough to truly judge our model. For example, consider that there are 98% samples of class A and 2% samples of class B in our training set. Then our model can easily get 98% training accuracy by simply predicting every training sample belonging to class A.

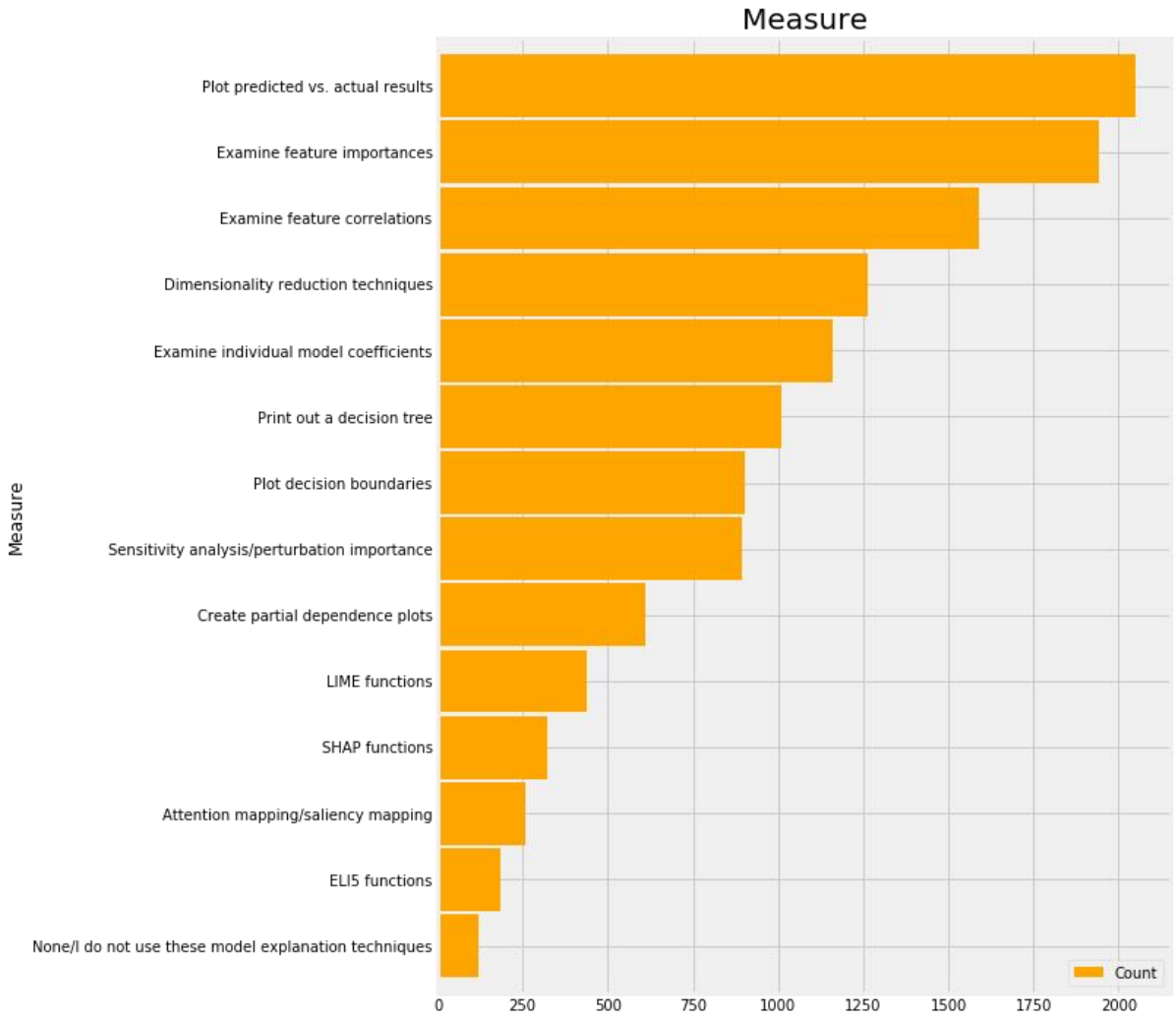
When the same model is tested on a test set with 60% samples of class A and 40% samples of class B, then the test accuracy would drop down to 60%. Accuracy is great, but gives us the false sense of achieving better model.So just reporting the accuracy to a non-technical person can prove to be a wrong perception for that person. Some of the better ways to explain your model's behaviour:

unfold_lessHide code

In [23]:

```
l1=[col for col in ds if col.startswith("What methods do you prefer for explaining and/or interpreting decisions that are made by ML models? (Select all that apply) - ")]
col1=[]
col2=[]
l2=ds[l1[:-2]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
measure=pd.DataFrame({'Measure':col1,'Count':col2}).sort_values(by='Count')
measure.set_index('Measure').plot.barh(width=0.95,color='orange')
plt.gcf().set_size_inches(8,12)
```

```
plt.title('Measure',size=20)

plt.show()
```



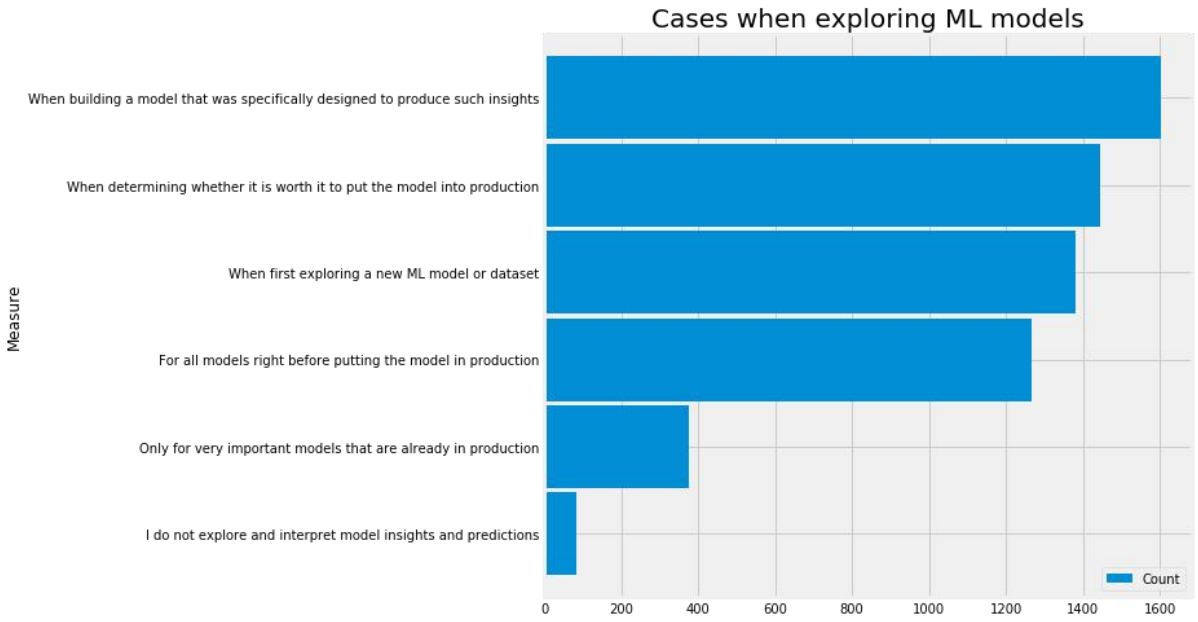
Plotting the predicted values vs the actual values gives a better idea about where and by how much margin is your model going wrong. For example, lets say we have a linear regression problem and we plot the predicted and actual values with a regressed diagonal line, we can see how far from the diagonal is our predicted values, using which we can tune our model to reduce the mean error. Similarly, feature importance gives us an idea about the most relevant features that contribute to the model's performance. Metrics like accuracy are one thing, but making sure that the model does something reasonable is important as well!

Data Scientists in most of the cases do explore the ML model insights for better stable model. But it really varies at what point of time in the ML pipeline do you do this. At the start, during model building/prototyping, during production or for every model,etc. Some of the common cases/conditions:

unfold_lessHide code

In [24]:

```
l1=[col for col in ds if col.startswith("In what circumstances would you explore model insights and interpret your model's predictions? (Select all that apply) - ")]
col1=[]
col2=[]
l2=ds[l1[:]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
measure=pd.DataFrame({'Measure':col1,'Count':col2}).sort_values(by='Count')
measure.set_index('Measure').plot.barh(width=0.95)
plt.gcf().set_size_inches(8,8)
plt.title('Cases when exploring ML models',size=20)
plt.show()
```



Another thing I would like to focus on is to make your code readable and reproducible. If your code is reproducible, others can also use them with minimal efforts and outputs can be studied in a better way. Reproducibilty is thus a very important aspect in Data Science. Have a look at this to understand why it is so important, and I believe other Data Scientists believe this:

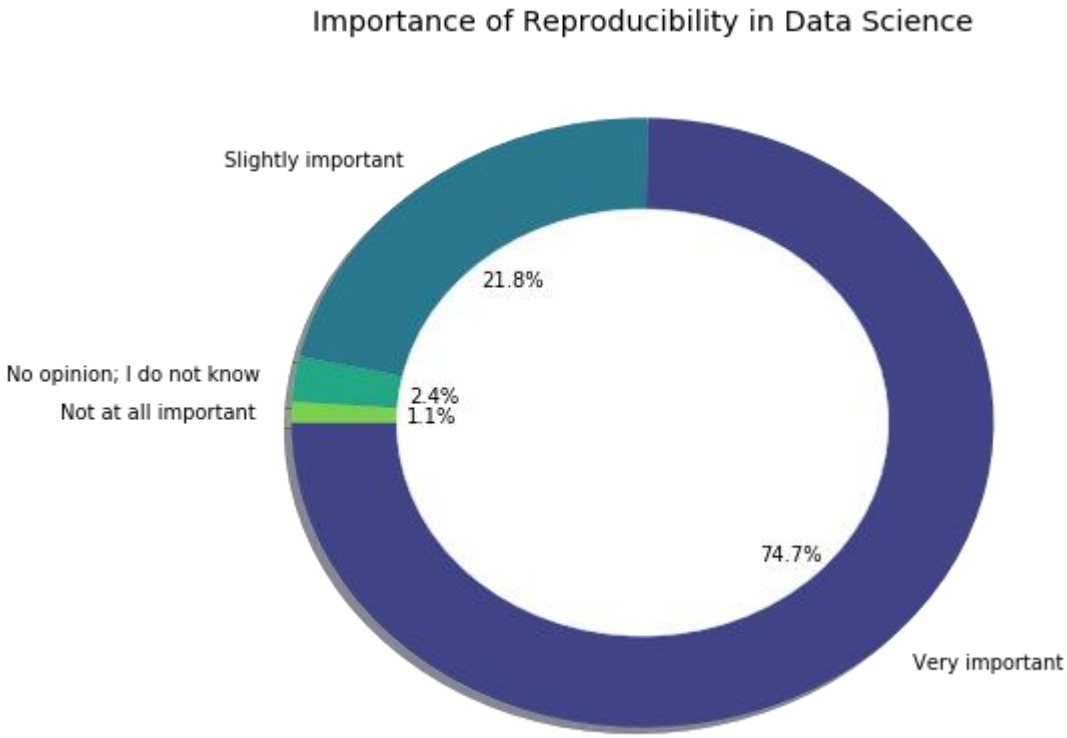
unfold_lessHide code

In [25]:

```
ds['How do you perceive the importance of the following topics? - Reproducibility in data science'].value_counts().plot.pie(autopct='%1.1f%%',shadow=True,startangle=180,colors=sns.color_palette('viridis',4))

plt.title('Importance of Reproducibility in Data Science')

my_circle=plt.Circle( (0,0), 0.7, color='white')

plt.gcf().set_size_inches(7,7)

plt.gca().add_artist(my_circle)

plt.ylabel('')
```

Out[25]:

Text(0,0.5,'')



About 95% of the Data Scientists do say that Reproducibility is important in Data Science.

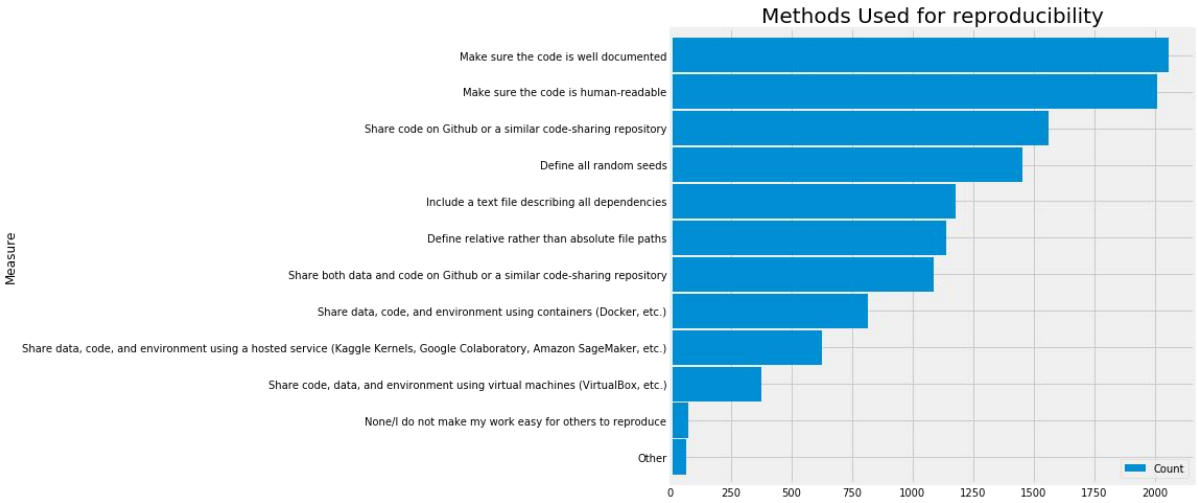
Some of the methods we use to do so:

unfold_lessHide code

In [26]:

```
l1=[col for col in ds if col.startswith("What tools and methods do you use to make your work easy to reproduce? (Select all that apply) - Selected Choice - ")]
col1=[]
col2=[]
l2=ds[l1[:]]
for i in l2.columns:
    col1.append(ds[i].value_counts().index.values[0])
    col2.append(ds[i].value_counts().values[0])
measure=pd.DataFrame({'Measure':col1,'Count':col2}).sort_values(by='Count')
measure.set_index('Measure').plot.barh(width=0.95)
plt.gcf().set_size_inches(8,8)
plt.title('Methods Used for reproducibility',size=20)
```

plt.show()



The top 2 methods viz well documented and human-readable code is very important irrespective of the domain of work. Anyone reading a well orgainsed piece of code can easily get a brisk idea of it even if he is from a non technical background. Sharing of code on public code-sharing repos like Github is another way of maintaining your code as well as showing your knowledge to potential recruiters. One thing that data scientists often fail to do is set the seed values for their analysis. This makes it impossible to exactly recreate machine learning studies. Many machine learning algorithms include a stochastic element and, while robust results might be statistically reproducible, there is nothing to compare with the warm glow of matching the exact numbers produced by someone else. If you are using scripts and source code control your seed values should be set in your scripts.

Q12. How much can I expect to earn?

Answer:

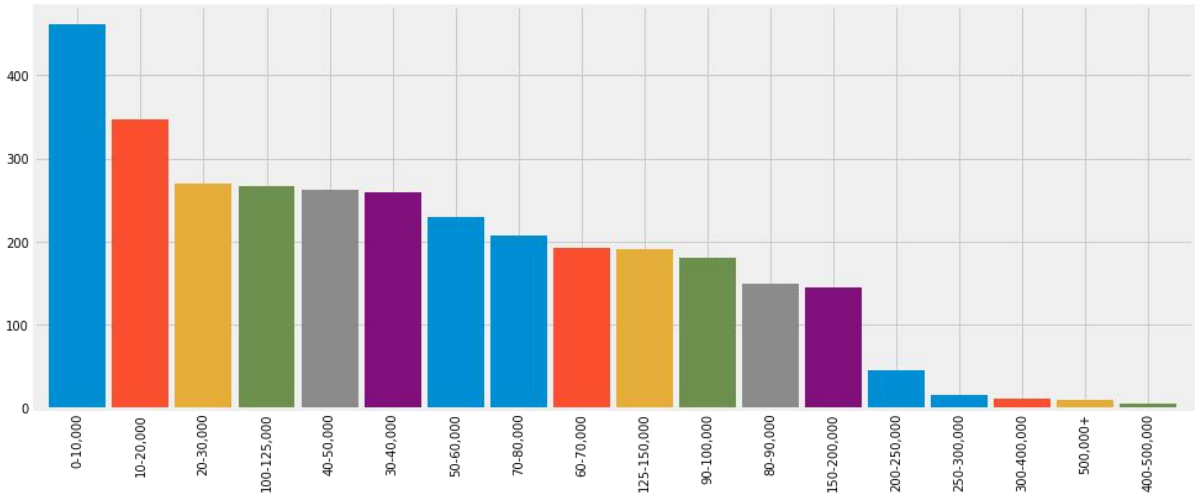
Great question indeed!! I was expecting this one :p. Data Scientists are one of the most well payed professionals in the industry. Have a look at the trend in DS salaries:

The question is simple, but the answer is not that simple. Your salary depends on a lot of factors, like your Country of residence, your work experience,etc. Some salary ranges reported by Data Scientists over the world:

unfold_lessHide code

In [27]:

```
ds['What is your current yearly compensation (approximate $USD)?'].value_counts()[1:].plot.bar(width=0.9)
plt.gcf().set_size_inches(15,6)
plt.show()
```



linkcode

You see!! There is a lot of variance in the compensations. According to the US salary reports, an entry level Data Scientist earns anywhere between 40-60000 dollars, a mid-level DS earns anywhere between 90-125000 dollars whereas a senior level DS earns more than 150000 dollars. This will surely change with the demographics. Some of the most important factors influencing a Data Scientist salary are:

unfold_lessHide code

In [28]:

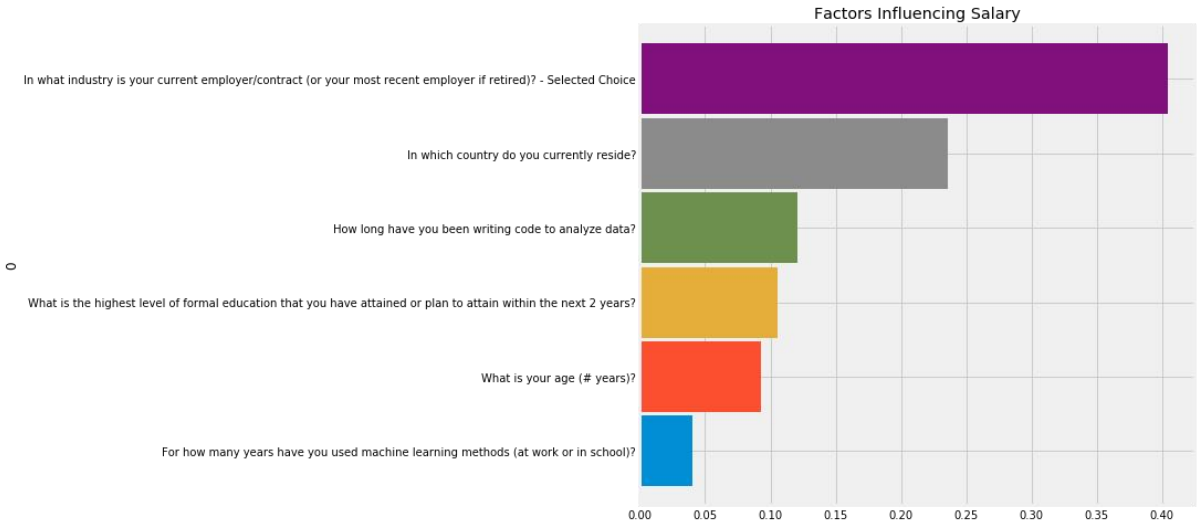
```
model=ds[['What is your age (# years)?','What is the highest level of formal education that you have attained or plan to attain within the next 2 years?','In which country do you currently reside?','In what industry is your current employer/contract (or your most recent employer if retired)? - Selected Choice','What is your current yearly compensation (approximate $USD)?','How long have you been writing code to analyze data?','For how many years have you used machine learning methods (at work or in school)?',]]

model.loc[np.logical_not(model['What is the highest level of formal education that you have attained or plan to attain within the next 2 years?'].isin(["Master's degree","Doctoral degree","Bachelor's degree"])),'What is the highest level of formal education that you have attained or plan to attain within the next 2 years?']='Others'

model.loc[(model['What is your age (# years)?'].isin(["18-21","22-24"])),'What is your age (# years)?']='<25'

model.loc[(model['What is your age (# years)?'].isin(["25-29","40-44","30-34","35-39"])),'What is your age (# years)?']='25-45'

model.loc[(model['What is your age (# years)?'].isin(["45-49","50-54","55-59","60-69","70-79",'80+'])),'What is your age (# years)?']='45+'

model.loc[model['In which country do you currently reside?'].isin(['France','Germany','Spain','Netherlands','Italy','Poland','Belgium','Portugal','Switzerland','United Kingdom of Great Britain and Northern Ireland']),'In which country do you currently reside?']='Europe'

model.loc[np.logical_not(model['In which country do you currently reside?'].isin(["United States of America","India","Europe"])),'In which country do you currently reside?']='Others'

model.loc[model['How long have you been writing code to analyze data?'].isin(['< 1 year','I have never written code but I want to learn','I have never written code and I do not want to learn']),'How long have you been writing code to analyze data?']='1'

model.loc[model['How long have you been writing code to analyze data?'].isin(['1-2 years','3-5 years','5-10 years']),'How long have you been writing code to analyze data?']='1-10'
```

```
model.loc[model['How long have you been writing code to analyze data?'].isin(['30-40 years','20-30 years','10-20 years','40+ years']),'How long have you been writing code to analyze data?']='10+'

model.loc[model['For how many years have you used machine learning methods (at work or in school)?'].isin(['I have never studied machine learning but plan to learn in the future','I have never studied machine learning and I do not plan to','< 1 year']),'For how many years have you used machine learning methods (at work or in school)?']='1'

model.loc[model['For how many years have you used machine learning methods (at work or in school)?'].isin(['1-2 years','2-3 years','3-4 years','5-10 years','4-5 years']),'For how many years have you used machine learning methods (at work or in school)?']='1-10'

model.loc[np.logical_not(model['For how many years have you used machine learning methods (at work or in school)?'].isin(['<1','1-10'])),'For how many years have you used machine learning methods (at work or in school)?']='10+'

model.loc[model['What is your current yearly compensation (approximate $USD)?'].isin(['0-10,000','10-20,000','20-30,000','40-50,000','30-40,000']),'What is your current yearly compensation (approximate $USD)?']='Low'

model.loc[model['What is your current yearly compensation (approximate $USD)?'].isin(['125-150,000','100-125,000','90-100,000','60-70,000','80-90,000','70-80,000','50-60,000']),'What is your current yearly compensation (approximate $USD)?']='Average'

model.loc[model['What is your current yearly compensation (approximate $USD)?'].isin(['150-200,000','200-250,000','250-300,000','300-400,000','500,000+','400-500,000']),'What is your current yearly compensation (approximate $USD)?']='High'

model.loc[np.logical_not(model['What is your current yearly compensation (approximate $USD)?'].isin(['Low','Average','High'])),'What is your current yearly compensation (approximate $USD)?']='Others'


from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import LabelEncoder

from sklearn.model_selection import train_test_split

from sklearn import metrics

le=LabelEncoder()

for i in model.columns:

    le=le.fit(model[i].astype(str))

    model[i]=le.transform(model[i].astype(str))


from sklearn.model_selection import cross_val_predict
```

```
X=model[model.columns.difference(['What is your current yearly compensation (approximate $USD)?'])]

Y=model['What is your current yearly compensation (approximate $USD)?']

model1=RandomForestClassifier(n_estimators=100,random_state=10)

model1.fit(X,Y)

pd.Series(model1.feature_importances_,X.columns).sort_values(ascending=True).plot.barh(width=0.95)

plt.gcf().set_size_inches(8,8)

plt.title('Factors Influencing Salary')
```

Out[28]:

Text(0.5,1,'Factors Influencing Salary')



I hope this AMA session proved to be a great foundation for many aspiring Data Scientists. It would like to thank you all for asking such eminent questions that can really help data science aspirants. Thanks for having me here!!