Untitled1

December 10, 2019

```
In [6]: from os import walk
        import json

        directory = "C:/Users/Chen/Desktop/big data/output"
        files = []
        for (dirpath, dirnames, filenames) in walk(directory):
            files.extend(filenames)
            break
        datas = {}
        datas["datasets"]=[]
        for file in files:
            filePath = directory + "/" + file
            with open(filePath) as json_file:
                dataset = json.load(json_file)
                datas["datasets"].append(dataset)
        y = json.dumps(datas, indent=2)
        output_file = directory + "/all_datasets.json"
        f = open(output_file, 'w')
        print(y, file=f)
        f.close()
```

```
In [47]: from os import walk
         import json
         import matplotlib.pyplot as plt

         directory = "D:/big data/output"
         files = []
         for (dirpath, dirnames, filenames) in walk(directory):
             files.extend(filenames)
             break
         col_count = 0
         int_count = 0
         real_count = 0
         date_count = 0
         text_count = 0
         for file in files:
             filePath = directory + "/" + file
             with open(filePath) as json_file:
```

```
        dataset = json.load(json_file)
        for col in dataset["columns"]:
            col_count += 1
            for t in col["data_types"]:
                if t["type"] == "INTEGER (LONG)":
                    int_count += 1
                if t["type"] == "REAL":
                    real_count += 1
                if t["type"] == "DATE/TIME":
                    date_count += 1
                if t["type"] == "TEXT":
                    text_count += 1

num_list = []
num_list.append(int_count / col_count)
num_list.append(real_count / col_count)
num_list.append(date_count / col_count)
num_list.append(text_count / col_count)
```

In [42]:

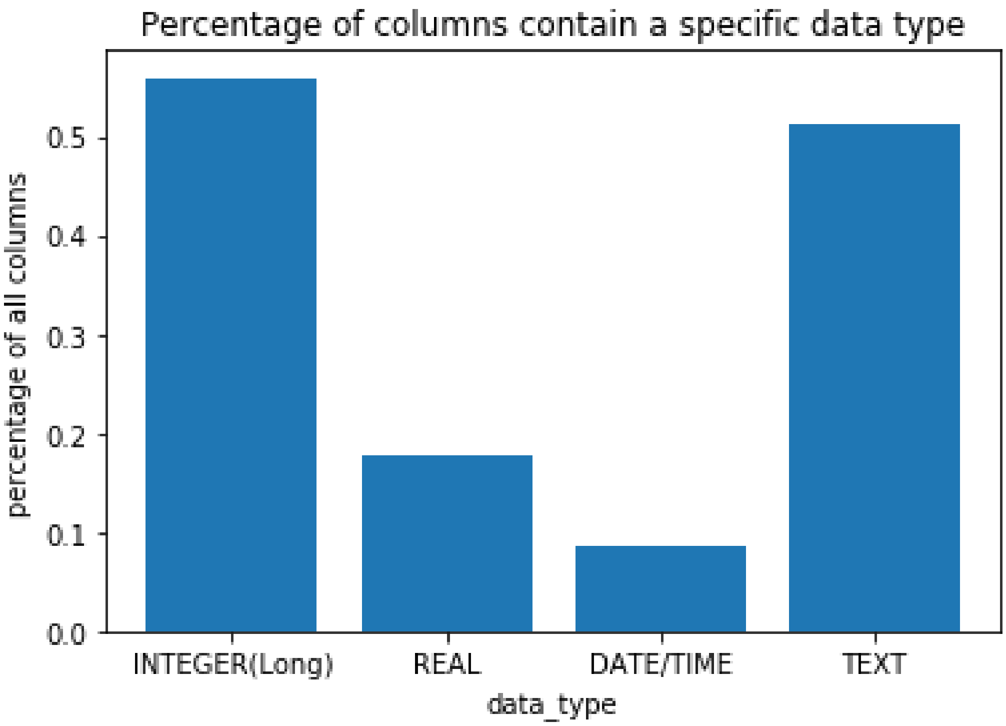
```
from os import walk
import json
import matplotlib.pyplot as plt

directory = "D:/big data/output"
files = []
for (dirpath, dirnames, filenames) in walk(directory):
    files.extend(filenames)
    break
col_count = 0
mis_count = 0
for file in files:
    filePath = directory + "/" + file
    with open(filePath) as json_file:
        dataset = json.load(json_file)
        for col in dataset["columns"]:
            col_count += 1
            for t in col["data_types"]:
                mis_count += 1
```

In [44]:

```
from os import walk
import json
import matplotlib.pyplot as plt

directory = "D:/big data/output"
files = []
for (dirpath, dirnames, filenames) in walk(directory):
    files.extend(filenames)
    break
```

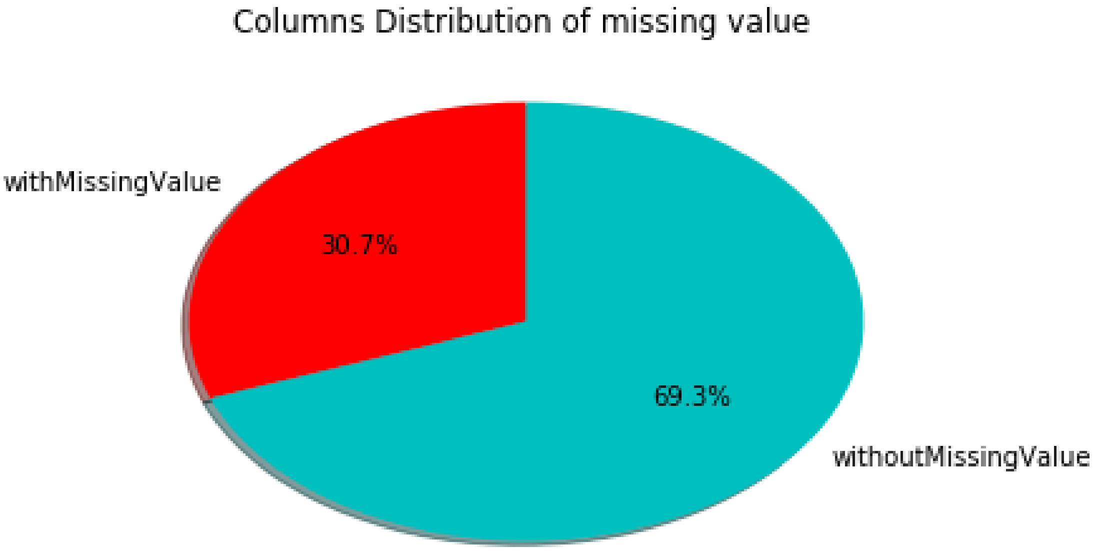
```
         col_count = 0
         heter_count = 0
         for file in files:
             filePath = directory + "/" + file
             with open(filePath) as json_file:
                 dataset = json.load(json_file)
                 for col in dataset["columns"]:
                     col_count += 1
                     if len(col["data_types"]) > 1:
                         heter_count += 1
```

```
In [31]: x=[1,2,3,4]
         LABELS = ["INTEGER(Long)", "REAL", "DATE/TIME", "TEXT"]
         plt.bar(x, num_list, align='center')
         plt.xticks(x, LABELS)
         plt.title('Percentage of columns contain a specific data type')
         plt.ylabel('percentage of all columns')
         plt.xlabel('data_type')
         plt.show()
```

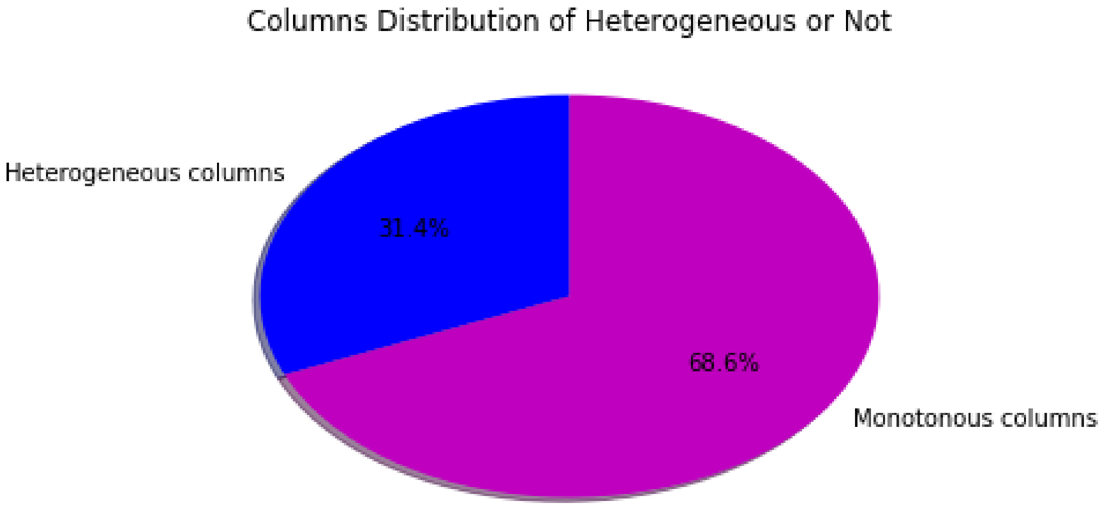


```
In [41]: slices = [mis_count, col_count - mis_count]
         activities = ['withMissingValue', 'withoutMissingValue']
         cols = ['r','c']
```

```
        plt.pie(slices,
                labels=activities,
                colors=cols,
                startangle=90,
                shadow= True,
                autopct='%1.1f%%')

        plt.title('Columns Distribution of missing value ')
        plt.show()
```



```
In [46]: slices = [heter_count, col_count - heter_count]
         activities = ['Heterogeneous columns', 'Monotonous columns']
         cols = ['b','m']

         plt.pie(slices,
                 labels=activities,
                 colors=cols,
                 startangle=90,
                 shadow= True,
                 autopct='%1.1f%%')

         plt.title('Columns Distribution of Heterogeneous or Not')
         plt.show()
```


Columns Distribution of Heterogeneous or Not
Heterogeneous columns
31.4%
68.6%
Monotonous columns