

detecting outliers using standard deviation

Aim = TO DETECT OUTLIERS

In []:

```
import numpy as np
import pandas as pd
import seaborn as sns
```

In []:

```
def generate_scores(mean=60,std_dev=12,num_samples=200):
    np.random.seed(27)
    scores = np.random.normal(loc=mean,scale=std_dev,size=num_samples)
    scores = np.round(scores,decimals=0)
    return scores
scores_data = generate_scores()
print( scores_data)
```

```
[75. 56. 67. 65. 63. 59. 58. 63. 70. 75. 67. 65. 53. 48. 75. 54. 57. 57.
 57. 54. 53. 66. 62. 55. 65. 45. 65. 29. 69. 49. 58. 81. 91. 58. 61. 65.
 50. 78. 65. 63. 72. 46. 73. 61. 60. 70. 42. 67. 82. 72. 74. 69. 58. 64.
 70. 65. 44. 90. 60. 58. 52. 52. 63. 59. 65. 56. 44. 58. 58. 45. 61. 42.
 51. 67. 61. 75. 72. 71. 58. 67. 60. 69. 62. 64. 52. 58. 75. 35. 43. 63.
 62. 67. 72. 72. 58. 64. 64. 53. 47. 45. 45. 67. 73. 78. 69. 66. 69. 59.
 57. 63. 66. 34. 45. 63. 65. 61. 47. 68. 53. 70. 73. 70. 38. 78. 62. 83.
 60. 67. 61. 57. 64. 63. 48. 61. 88. 84. 62. 67. 39. 56. 72. 66. 62. 67.
 77. 60. 46. 59. 65. 59. 54. 79. 38. 46. 50. 72. 61. 47. 65. 29. 71. 59.
 65. 46. 56. 67. 62. 64. 57. 76. 64. 64. 73. 62. 61. 38. 20. 55. 63. 55.
 39. 51. 70. 58. 43. 54. 82. 65. 67. 50. 56. 54. 72. 54. 42. 76. 67. 74.
 98. 53.]
```

In []:

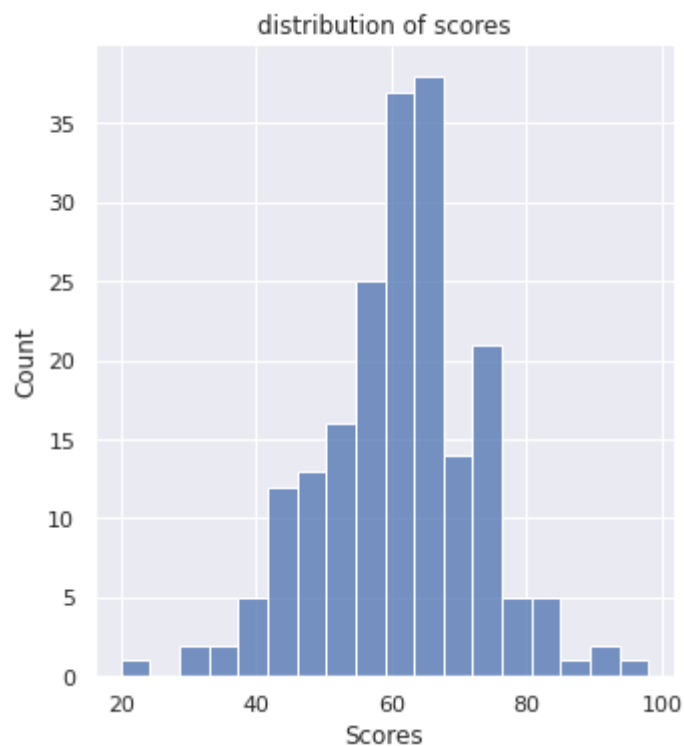
```
scores.info()
```

In []:

```
sns.set_theme()  
sns.displot(data=scores_data).set(title="distribution of scores",xlabel="Scores")
```

Out[3]:

<seaborn.axisgrid.FacetGrid at 0x7fe8df1f4fa0>



In []:

```
df_scores=pd.DataFrame(scores_data,columns=["score"])  
df_scores.head()
```

Out[18]:

	score
0	75.0
1	56.0
2	67.0
3	65.0
4	63.0

In []:

```
df_scores.mean()
```

Out[11]:

```
score    61.005  
dtype: float64
```

In []:

```
df_scores.std()
```

Out[12]:

```
score    11.854434
dtype: float64
```

In []:

```
lower_limit=df_scores.mean() -3*df_scores.std()
upper_limit=df_scores.mean() +3*df_scores.std()
print(lower_limit)
print(upper_limit)
```

```
score    25.441697
dtype: float64
score    96.568303
dtype: float64
```

In []:

```
df_scores_filtered=df_scores[(df_scores[['score']]>lower_limit)&(df_scores[['score']]<up
df_scores_filtered
```

Out[19]:

	score
0	75.0
1	56.0
2	67.0
3	65.0
4	63.0
...	...
195	76.0
196	67.0
197	74.0
198	NaN
199	53.0

200 rows × 1 columns

detecting outliers using z score

In []:

```
df_scores['z_score']=df_scores[(df_scores[['score']]>lower_limit)&(df_scores[['score']]<
df_scores_filtered
```

Out[20]:

	score
0	75.0
1	56.0
2	67.0
3	65.0
4	63.0
...	...
195	76.0
196	67.0
197	74.0
198	NaN
199	53.0

200 rows × 1 columns

Conclusion = Detectd Ouliers by using Stadard Deviation .