

La distribution de TLOC montre une inclinaison vers la droite, ce qui suggère que bien que la majorité des classes présentent un nombre de lignes relativement bas, il existe un certain nombre de classes avec un nombre très élevé de lignes. Ces classes atypiques ont pour effet d'augmenter considérablement la moyenne globale.

Voici les résultats pour tloc :

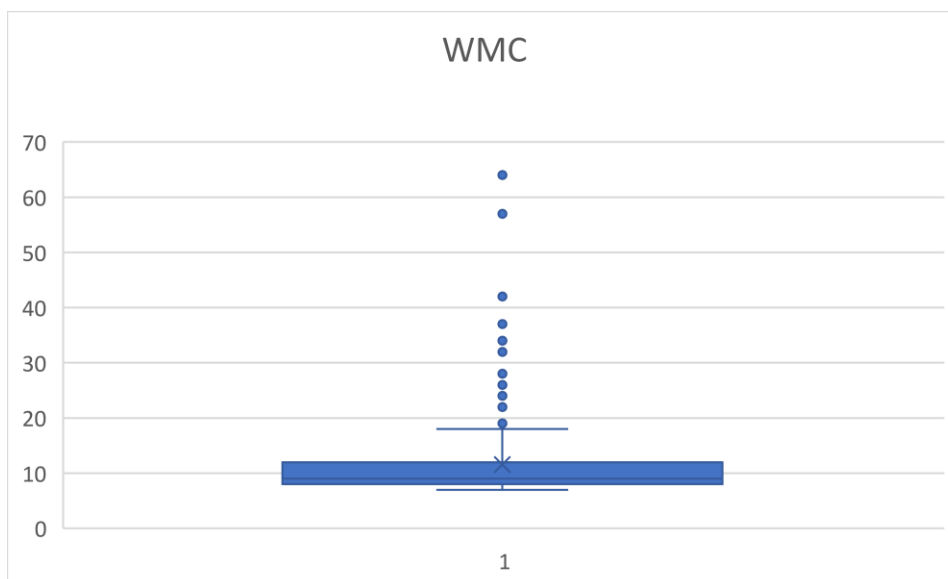
Moyenne : 115.13

Médiane : 83.00

Premier Quartile : 47.50

Troisième Quartile : 124.50

Limite Supérieure : 240.00



La distribution de WMC, bien que moins prononcée que celle de TLOC, révèle toujours une certaine variabilité. Elle indique que, si un grand nombre de classes possèdent un nombre modéré de méthodes, quelques-unes se distinguent par un nombre nettement plus élevé de méthodes, contribuant ainsi à l'hétérogénéité observée dans cette métrique.

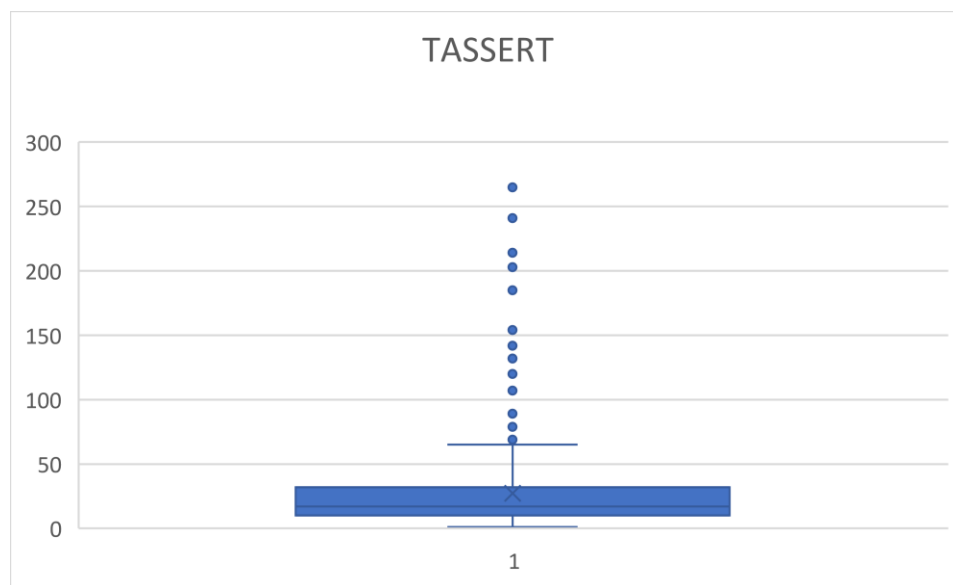
Moyenne : 11.58

Médiane : 9.00

Premier Quartile : 8.00

Troisième Quartile : 12.00

Limite Supérieure : 18.00



La distribution de TASSERT montre une médiane plus basse et la présence de valeurs extrêmes, indiquant que, bien que la plupart des classes aient un nombre inférieur d'assertions, quelques-unes ont un nombre beaucoup plus élevé d'assertions.

Moyenne : 27.19

Médiane : 17.00

Premier Quartile : 10.00

Troisième Quartile : 32.00

Code python :

```

import pandas as pd

def calculer_stats(df, colonnes):
    resultats = {}

    for col in colonnes:
        moyenne = df[col].mean()
        mediane = df[col].median()
        q1 = df[col].quantile(0.25)
        q3 = df[col].quantile(0.75)
        iqr = q3 - q1
        limite_sup = q3 + 1.5 * iqr
        resultats[col] = {
            'Moyenne': moyenne,
            'Mediane': mediane,
            '1er Quartile': q1,
            '3eme Quartile': q3,
            'Limite Superieure': limite_sup
        }

    return resultats

chemin_fichier = '/jfreechart-test-stats.csv'
donnees = pd.read_csv(chemin_fichier)
donnees.columns = donnees.columns.str.strip()
# Métriques à calculer
colonnes = ['TLOC', 'WMC', 'TASSERT']

# Calcul des statistiques
resultats_stats = calculer_stats(donnees, colonnes)
print(resultats_stats)

```