

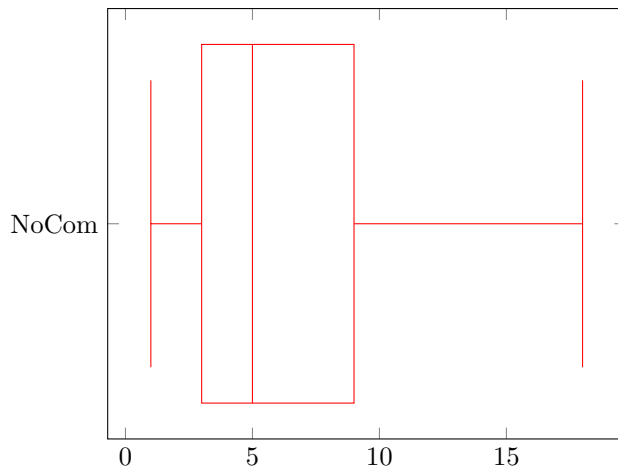
IFT-3913 TP3

Laurent Charlebois et Nisrine Mastas

18 novembre 2022

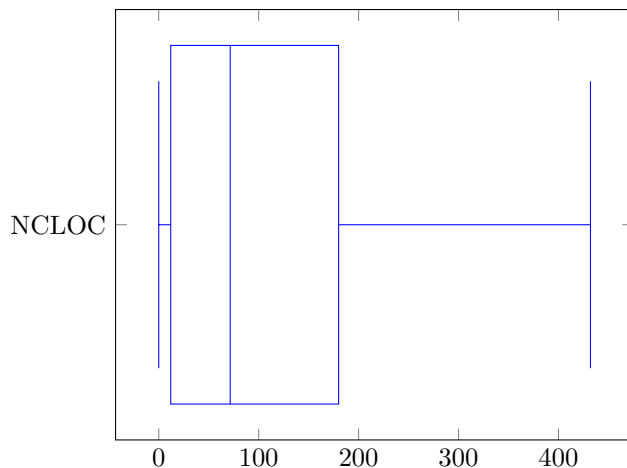
1 Q1

Pour la métrique NoCom nous obtenons la boîte à moustache suivante



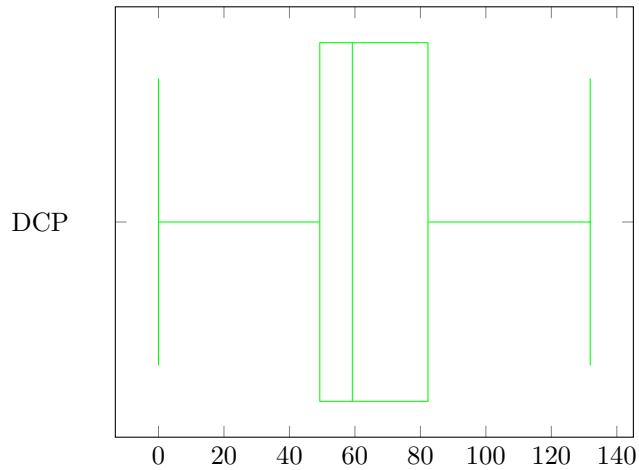
Ce qui démontre que la plupart des classes on entre 3 et 9 commit et une classes devrait avoir 18 commit au maximum. Par contre, nous avons 19 classe qui dépassent ce maximum (voir Q1ExtremePointNames.txt pour voir les noms de ces classes). Cela démontre qu'il y a quelque classes qui ont été beaucoup plus modifié que les autres. On remarque que le minimum est 1 et non 0, car il impossible d'avoir une classes avec 0 commit.

Pour la métrique NCLOC, nous obtenons la boîte



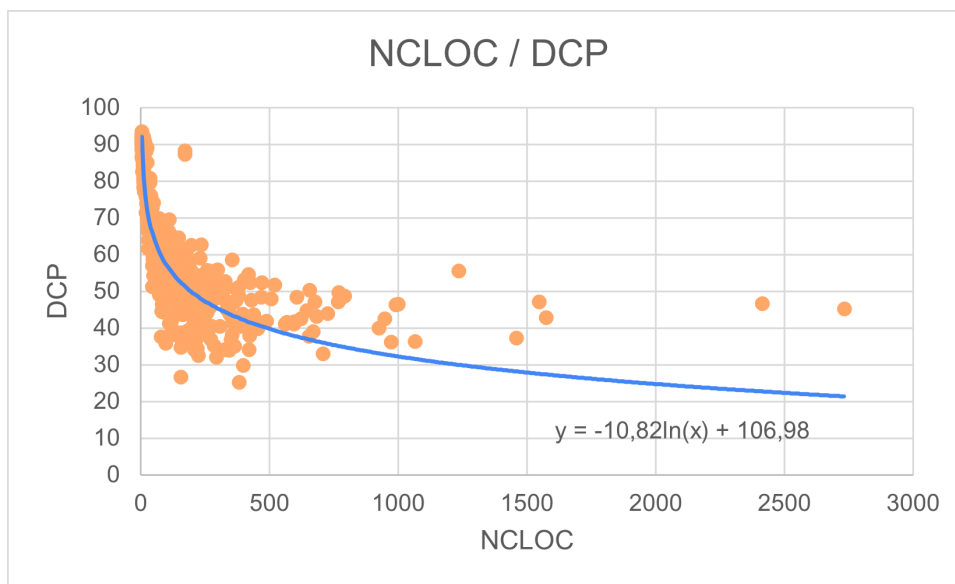
Ce qui démontre que la plupart des classes on entre 12 et 180 lignes de code et une classe devrait avoir 432 lignes de code au maximum. Par contre, nous avons 39 classes qui dépassent ce maximum. Cela démontre qu'il y a plusieurs classes qui ont beaucoup de lignes de codes et qui devrait probablement être réduite.

Pour la métrique DCP, nous obtenons la boîte



Ce qui démontre que la plupart des classes ont une densité de commentaire entre 49.27 et 82.31 et une classe ne devrait pas avoir une densité de commentaire de plus de 131.87. Cela n'est pas le cas pour aucune des classes. Nous avons donc une bonne densité de commentaire standard pour toutes les classes.

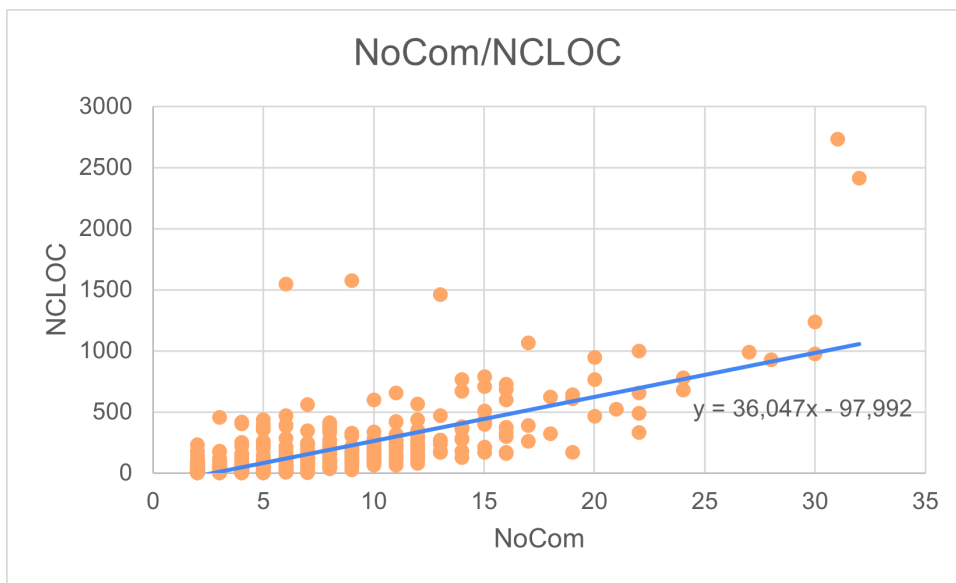
2 Q2



La droite de régression : $y = -10,82\ln(x) + 106,98$

Le coefficient de corrélation NCLOC et DCP est : -0.5076314496494513

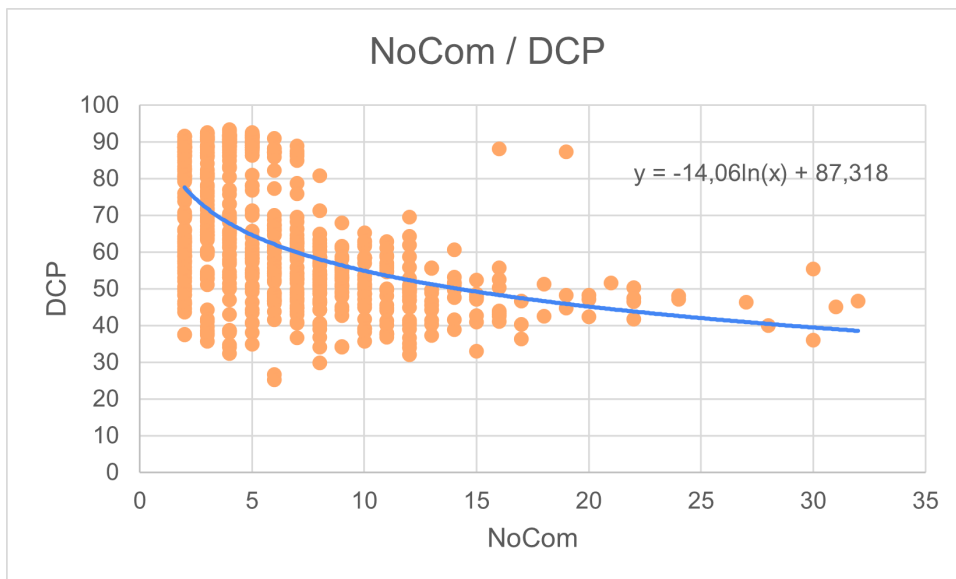
Le coefficient de corrélation est négatif donc NCLOC et DCP ont une corrélation linéaire négative



La droite de régression : $y = 36,047x - 97,992$

Le coefficients de corrélation NOCom et NCLOC est : 0.7145249277762028

Le coefficients de corrélation est proche de 1 donc NOCom et NCLOC ont une bonne corrélation linéaire positive



La droite de régression : $y = -14,06 \ln(x) + 87,318$

Le coefficients de corrélation NOCom et DCP est : -0.48777520610094754

Le coefficients de corrélation est negatif donc NOCom et DCP ont une corrélation linéaire négative

3 Q3

3.1 Choix d'étude

Nous avons choisi de faire une étude de cas, car nous avons déjà des données précises collecter par les métrique et nous avons seulement une entité. Donc, il est impossible de faire une expérience et nous avons déjà plus de données que ce qu'un sondage nous donnerais.

3.2 Énoncé des hypothèses

L'Hypothèse est « *les classes qui ont été modifiées plus de 10 fois sont mieux commentées que celles qui ont été modifiées moins de 10 fois* »

3.3 Définition des variables

Nous avons la variable d'état NoCom, le nombre de commit. Mais plus précisément si NoCom est plus grand que 10 ou NoCom est plus petit que 10. Cela sont nos deux catégories de classes qui sont interprétées.

Nous avons la variable dépendantes DCP, qui indique la densité de commentaires de chaque classes.

3.4 Interprétation et généralisation des résultats

Nous avons comme résultat que les fonctions qui ont plus que 10 commit ont une densité de commentaire en moyenne de : 48.19, avec une médiane de 47.58 .

Pour que des fonctions qui ont moins que 10 commit on a une densité de commentaire en moyenne de : 67.52, avec une médiane de 64.61 .

Nous constatons, que les classes avec moins de 10 commit on une densité de commentaires plus grande que les classes qui ont plus que 10 commit. Cela indiquerait que les classes modifié moins de 10 fois sont mieux commentées que celles qui ont été modifiées plus que 10 fois.

Cela réfute notre Hypothèse de départ.

3.5 Discussion des menaces à la validité

Par contre, il a plusieurs points qui pourrait biaiser nos calcul et notre conclusion.

Tout d'abord, la densité de commentaire n'est pas nécessairement une mesure de qualité des commentaires et donc il ne faut pas tiré une conclusion définitive avec seulement cette métrique. Pour mesurer la qualité de commentaire, qui est quelque chose de plus subjectif, il faudrait faire une expérience avec plusieurs personnes qui analyse les commentaires.

Ensuite, les classes avec peu de commit sont souvent des classes plus petites que les fonctions avec plus de commit. Une classe plus petite à tendance à avoir une plus grande densité de commentaires, car il y a souvent moins de fonctions triviales qui ne nécessite pas de commentaires. Cela est un biais clair contre les classes qui sont modifiées plus de 10 fois.

Finalement, les classes modifié plus de 10 fois sont souvent des classes qui sont plus optimisé, donc nécessite moins de commentaires. De façon similaire au point précédent, cela crée un biais contre les fonctions modifié plus de 10 fois.