

El Impacto de la Cantidad de Líneas de Código sobre las Pruebas de Caja Negra

Eduardo E. Martínez, Gabriel Cepeda

Departamento de Ingeniería en Ciencias de la Computación, PUCMM
Santiago de los Caballeros, República Dominicana

RESUMEN

Este estudio investiga la relación entre el método de prueba de Caja Negra y el desempeño de seis proyectos de Python desarrollados por estudiantes universitarios. La investigación se enfoca en analizar si la cantidad de líneas de código de los proyectos influye en su rendimiento al someterse a la prueba de caja negra.

La metodología de investigación se basó en un enfoque cuantitativo, utilizando estadística descriptiva para analizar los datos. Se evaluaron las puntuaciones y la cantidad de líneas de código de cada proyecto para identificar posibles correlaciones. Además, se realizó un análisis de regresión para explorar estadísticamente la relación entre la cantidad de líneas de código y la efectividad en la prueba de caja negra.

Los resultados revelaron que la mayoría de los proyectos obtuvieron puntuaciones moderadas en la prueba de Caja Negra. Sin embargo, no se encontró una relación estadísticamente significativa entre la cantidad de líneas de código y la puntuación en la prueba. Esto sugiere que el éxito en la prueba de Caja Negra está influenciado por otros factores más allá de la extensión del código.

Palabras Clave: Prueba de Caja Negra, prueba de software, Python, cantidad de líneas de código, calidad del software.

ABSTRACT

This study investigates the relationship between the Black Box testing method and the performance of six Python projects developed by university students. The research aims to analyze whether the number of lines of code in the projects influences their performance when subjected to Black Box testing.

The research methodology is based on a quantitative approach, using descriptive statistics to analyze the data. Scores and the number of lines of code in each project were evaluated to identify possible correlations. Additionally, a regression analysis was conducted to statistically explore the relationship between the number of lines of code and the effectiveness of the Black Box testing.

The results revealed that the majority of projects achieved moderate scores in the Black Box testing. However, no statistically significant relationship was found between the number of lines of code and the test scores. This suggests that the success in Black Box testing is influenced by other factors beyond the code's length.

Keywords: Black Box Testing, software testing, Python, number of lines of code, software quality.

I. INTRODUCCIÓN

A. Antecedentes

En esta era digital, el software ha pasado a formar una parte esencial del desarrollo de incontables actividades, lo que ha generado una constante y continua demanda de software de alta calidad. Como respuesta a esto, nacen las metodologías de prueba de software, entre las cuales destaca el *Black Box Testing*, o Prueba de Caja Negra. Esta metodología ha ganado una inmensa popularidad debido a su enfoque ágil y eficiente basado en proporcionar entradas y observar las salidas para garantizar que el software funcione de acuerdo a sus especificaciones (Verma, Khatana, 2017).

El funcionamiento del método de prueba de Caja Negra se fundamenta en una serie de pruebas que permiten evaluar el software sin depender de su implementación interna, lo que la hace una herramienta valiosa para evaluar la calidad del software de manera eficiente y objetiva (Liu, Tan, 2009).

B. Planteamiento de Problema

A la luz de sus números beneficios, se ha identificado una excelente oportunidad para aplicar el método de prueba de caja negra en entornos educativos, específicamente en la revisión de programas o proyectos asignados a estudiantes. Para probar la viabilidad de esta idea, se propone evaluar la calidad del código de proyectos finales de seis estudiantes de secundaria, todos desarrollados en el lenguaje de programación Python, tomando como referencia de calidad su desempeño al aplicárseles esta metodología de prueba.

Para llevar a cabo esta investigación, se ha formulado la siguiente pregunta orientadora: "*¿Existe alguna relación entre el número de líneas de código en un programa y su efectividad al someterse a la prueba de caja negra?*" En otras palabras, se busca investigar si la cantidad de líneas código presenta alguna relación o impacto sobre el desempeño del software en las pruebas de caja negra.

C. Revisión de la Literatura

El tema de las pruebas de software como método de aseguramiento de la calidad ha tenido una gran importancia en la última década, pues incluso se ha demostrado que estas son esenciales para construir un software robusto y de calidad (Bhasin, Khanna, Sudha, 2014). De la misma forma, existen estudios que plantean que los beneficios de estas pruebas van más allá de la calidad, sino

que también tienen influencias económicas, pues los resultados de estas pruebas definirán qué tantos recursos serán necesarios para mantener dicho software en el futuro. (Dustin, Garrett. 2009).

De la misma forma, este tema será cada vez más relevante en el futuro, por lo que es preciso resaltar que la presente investigación tendrá una utilidad significativa para nuevas investigaciones en el área de aseguramiento de la calidad de software mediante metodologías de prueba. Los resultados arrojados por esta investigación pueden servir de modelo para guiar futuras investigaciones, e indicará su efectividad para evaluar la calidad del código con respecto a una nueva medida: su longitud en términos de líneas de código. (Hamer, López, Jenkins, 2021).

II. METODOLOGÍA DE LA INVESTIGACIÓN

La estrategia de análisis de datos utilizada para responder a la pregunta de investigación planteada en este estudio estuvo basada en su totalidad en un enfoque cuantitativo (Dias, Katipearachchi, Sachethana, 2023). Para iniciar este proceso, se identificó una población objetivo compuesta por estudiantes de una escuela secundaria que han desarrollado proyectos finales en Python, partiendo de lo cual, para la muestra, se consideró utilizar un muestreo probabilístico estratificado, donde se dividiera la población en estratos homogéneos basados en el nivel escolar (curso) de los estudiantes. Sin embargo, debido al enfoque de esta investigación en seis proyectos específicos, se decidió, en su lugar, a utilizar un muestreo no probabilístico intencional o por conveniencia.

La metodología experimental general del estudio, en primera instancia, consistió en calcular las cantidades de líneas de código y en someter los seis proyectos finales evaluados a cinco pruebas de caja negra específicas para cada uno de ellos. Para esto, se utilizó el Entorno Integrado de Desarrollo (IDE) Visual Studio Code como instrumento principal. Cada una de las pruebas específicas consistieron en aplicar una serie de entradas a los programas, probando sus capacidades y evaluando si su comportamiento era el esperado o no. Como resultado de estas pruebas, se le asigna un punto en el caso de que el comportamiento del mismo sea el esperado. Por lo tanto, a cada proyecto se le daba la posibilidad de obtener una puntuación acumulativa de 0 a 5 dependiendo de su desempeño.

Finalmente, con los resultados obtenidos, se llevó a cabo un análisis de regresión lineal para modelar y evaluar si existe una relación entre la cantidad de líneas de código y las puntuaciones obtenidas por los proyectos según las pruebas de caja negra (Barb, Neill, Sangwan, Piovoso, 2014).

III. RESULTADOS

Dado el enfoque de nuestra investigación, el análisis cuantitativo de los resultados será llevado a cabo mediante métodos de estadística descriptiva que permitan extraer conclusiones fundamentadas de resultados numéricos. Para iniciar este proceso de análisis de los resultados obtenidos, se calcularon las medidas de tendencia central y de variabilidad para obtener una mejor comprensión general de los resultados.

Tabla I. Medidas de Tendencia Central

PROYECTOS	PUNTUACIÓN	CANT. DE LINEAS
TIC-TAC-TOE	4	53
MASTERMIND	3	40
CONNECT4	0	73
BATTLESHIP	4	96
CARRERA DE BOLIS	3	44
TIMBIRICHE	3	136
PROMEDIO	2.833	72.670
MEDIANA	3.000	48.500
MODA	3.000	44.000
DESV. ESTÁNDAR	1.630	38.820

Con estos resultados, se observó que la mayoría de los proyectos obtuvieron puntuaciones moderadas, con un promedio de 2.833 y una mediana de 3.000. El proyecto "CONNECT4" fue el único que obtuvo una puntuación de 0, indicando un desempeño deficiente. Por su parte, la cantidad de líneas de código varió significativamente entre los proyectos, con una desviación estándar de 38.820 partiendo de un promedio de 72.670 líneas.

Ya analizados los resultados individuales, procedimos a explorar la relación entre la cantidad de líneas de código y el desempeño en las prueba de caja negra mediante un análisis de regresión lineal. Todo el proceso de desarrollo del modelo de regresión lineal a utilizar será llevado a cabo en el lenguaje R, haciendo uso de funciones como *lm* para calcular el mismo.

Una vez completado, este modelo nos permitirá determinar estadísticamente si existe una relación significativa entre estas variables mediante el planteamiento de una hipótesis nula (H_0) inicial: *Existe relación significativa entre la cantidad de líneas de código y la puntuación en la prueba de caja negra.*

Ilustración 1. Resultados de análisis de regresión

```
modelo <- lm(puntuacion ~ lineas_codigo, data=df)
summary(modelo)
```

Call:
lm(formula = puntuacion ~ lineas_codigo, data = df)

Residuals:

1	2	3	4	5	6
1.1778	0.1848	-2.8330	1.1547	0.1826	0.1332

Coefficients:

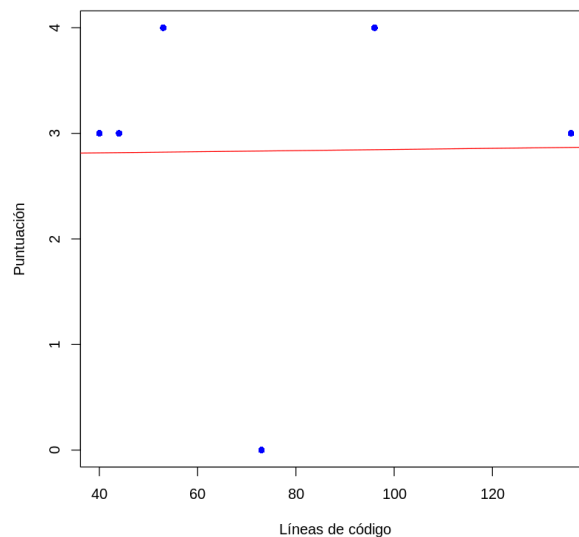
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.7937585	1.6137823	1.731	0.158
lineas_codigo	0.0005372	0.0199182	0.027	0.980

Residual standard error: 1.646 on 4 degrees of freedom
Multiple R-squared: 0.0001818, Adjusted R-squared: -0.2498
F-statistic: 0.0007274 on 1 and 4 DF, p-value: 0.9798

Como se observa, el modelo de regresión lineal utilizado arrojó un coeficiente para las líneas de código fue 0.0005, con un valor p de 0.980, lo que indica que no hay una relación estadísticamente significativa entre ambas variables. Además, el R-cuadrado múltiple fue de 0.00018, lo que sugiere que el modelo no explica de manera significativa la variabilidad en las puntuaciones de los proyectos.

Para representar la relación entre la variable *lineas de código* y *puntuación* en el conjunto de datos, se realizó el siguiente gráfico, el cual el eje X corresponde a la cantidad de líneas de código y el eje Y a la puntuación. Los puntos azules representan la intersección de los datos con los ejes, mientras que la línea roja indica la tendencia general de la relación lineal entre ambas variables.

Ilustración 2. Gráfico de dispersión y regresión lineal



IV. DISCUSIÓN

A. Conclusiones

Basados en los resultados individuales observados en la tabla 1, en cuanto a las puntuaciones podemos concluir que la mayoría de los programas desarrollados por los estudiantes de secundaria obtuvieron puntuaciones moderadas al someterse a la prueba de Caja Negra, reflejando un rendimiento promedio cercano a 3 sobre un total de 5 puntos posibles. Sin embargo, la significativa variabilidad demostrada por el valor de desviación estándar en las puntuaciones sugiere que algunos proyectos destacaron con puntajes muy altos o bajos en comparación con la media. Por

otro lado, los resultados obtenidos en cuanto a las líneas de código nos permiten concluir que la mayoría de los programas presentan una extensión relativamente concisa, dado que la mediana y la moda son inferiores al promedio.

Ahora, basados en los resultados obtenidos por el modelo de regresión lineal en la ilustración 1, podemos concluir que no existe una relación estadísticamente significativa entre la cantidad de líneas de código y su puntuación en la prueba de caja negra, ya que todos sus valores-p son mayores que 0.05. Por lo tanto, se rechaza la hipótesis nula que planteaba que existe relación significativa entre la cantidad de líneas de código y su puntuación en la prueba de caja negra.

La ilustración 2 respalda esta conclusión, pues muestra una amplia variabilidad entre los valores obtenidos por cada programa en ambas variables, lo que demuestra que algunos programas con una gran cantidad de líneas de código obtienen puntajes muy diferentes en las pruebas, mientras que otros con una cantidad similar de líneas pueden tener resultados muy distintos. En consecuencia, no se evidencia una relación clara entre estas variables.

Por lo tanto, dando respuesta a nuestra pregunta de investigación: “*¿Existe alguna relación entre el número de líneas de código en un programa y su efectividad al someterse a la prueba de caja negra?*”, podemos concluir que no existe una relación significativa entre la cantidad de líneas de código y el desempeño de los programas en la prueba de Caja Negra. Esta conclusión sugiere que el éxito en la prueba de Caja Negra está más vinculado a otros factores que sobrepasan el alcance de este estudio de investigación, en lugar de sólo la cantidad de líneas de código de los mismos.

B. Limitaciones

Aunque este estudio proporciona información valiosa sobre la relación entre la cantidad de líneas de código de los proyectos, y su desempeño en la prueba de Caja Negra, es importante reconocer que tuvo ciertas limitaciones en su alcance. Dentro de estas, la principal limitación a resaltar fue el tamaño reducido de la muestra, la cual consistió únicamente de seis proyectos de Python desarrollados por estudiantes de secundaria. Esta muestra limitada puede afectar la generalización de los resultados obtenidos, ya que no representa completamente la diversidad y complejidad que podrían encontrarse en un conjunto más amplio de proyectos. Igualmente, al tener una muestra tan pequeña, se observa una reducción en la precisión de los métodos de estudio y análisis empleados, lo que podría terminar por no revelar patrones o tendencias más sutiles en el conjunto de datos.

C. Recomendaciones

Para investigaciones futuras, se recomienda encarecidamente aumentar el tamaño de la muestra de proyectos de programación, lo que permitiría obtener resultados más precisos y representativos, fortaleciendo así la validez de los hallazgos de la investigación. Con una muestra más grande y diversa, se podría explorar una mayor variedad de enfoques de programación, soluciones y niveles de habilidad de los estudiantes, lo que proporcionaría una comprensión más completa de la relación entre la cantidad de líneas de código y el desempeño de los mismos en la prueba de caja negra. Además, una muestra más grande permitiría modificar el método de muestreo, de uno no probabilístico a uno probabilístico estratificado, para así poder realizar análisis comparativos más robustos entre los diferentes grupos de estudiantes, niveles educativos o entornos de aprendizaje.

V. REFERENCIAS BIBLIOGRÁFICAS

- H. Bhasin, E. Khanna, Sudha (2014), “Black Box Testing based on Requirement Analysis and Design Specifications”.** Recuperado de:
<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=5aa16644575c12228bbdd08e2088856ca433e810>
- A. Verma, A. Khatana, S. Chaudhary (2017), “A Comparative Study of Black Box Testing and White Box Testing”.** Recuperado de: <https://doi.org/10.26438/ijcse/v5i12.301304>
- H. Liu, H. Beng Kuan Tan (2009), “Covering code behavior on input validation in functional testing”.** Recuperado de: <https://doi.org/10.1016/j.infsof.2008.07.001>
- S. Hamer, C. Quesada-López, M. Jenkins (2021), “Students Projects’ Source Code Changes Impact on Software Quality Through Static Analysis”.** Recuperado de:
https://doi.org/10.1007/978-3-030-85347-1_39
- E. Dustin, T. Garrett, B. Gauf (2009), “Implementing Automated Software Testing: How to Save Time and Lower Costs While Raising Quality”.** Recuperado de:
<https://www.oreilly.com/library/view/implementing-automated-software/9780321619600/>
- D. De Silva, T. Dias, M. Katipearachchi (2023), “The Relationship between Code Complexity and Software Quality: An Empirical Study”.** Recuperado de:
https://www.researchgate.net/publication/370761578_The_Relationship_between_Code_Complexity_and_Software_Quality_An_Empirical_Study
- A. Barb, C. Neill, R. Sangwan. (2014), “A statistical study of the relevance of lines of code measures in software projects”.** Recuperado de: <https://doi.org/10.1007/s11334-014-0231-5>