

Q-Q Report

Statistical significance concerning the *speed of retrieval* for table counts for billing, historical, and current data in our datasets was tested. Our null hypothesis (H_0) is that the average indexed field for billing, historical, and current items in our datasets makes record retrieval significantly faster than non-indexed record retrievals using 'select count (*)' versus 'select count indexed field' on the ITV_BASE, OHI_BASE, and OCI_BASE tables. Our alternative hypothesis (H_a) is that indexed field retrieval by way of select statements is NOT significantly different than non-indexed record retrievals. In order to either accept or reject the null hypothesis, we use plotting based on small datasets from the respective tables.

Correspondingly, the Pearson product-moment correlation is a parametric measure of *association for two variables*. It measures both the strength and the direction of a linear relationship. If one variable X is an exact linear function of another variable Y , a positive relationship exists if the correlation is 1 and a negative relationship exists if the correlation is -1. If there is no linear predictability between the two variables, the correlation is 0. If the two variables are normal with a correlation 0, the two variables are independent. However, correlation does not imply causality because, in some cases, an underlying causal relationship might not exist.

TABLE 1

Pearson's Product-moment Correlation

Pearson's product-moment correlation

data: itv\$index and itv\$noIndex
t = 9.1372, df = 3, p-value = 0.002771
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval: 0.7526987 0.9988966
sample estimates: cor 0.9825035

Pearson's product-moment correlation

data: ohi\$index and ohi\$noIndex
t = 8.0525, df = 3, p-value = 0.004
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval: 0.6938263 0.9985866
sample estimates: cor 0.9776403

Pearson's product-moment correlation

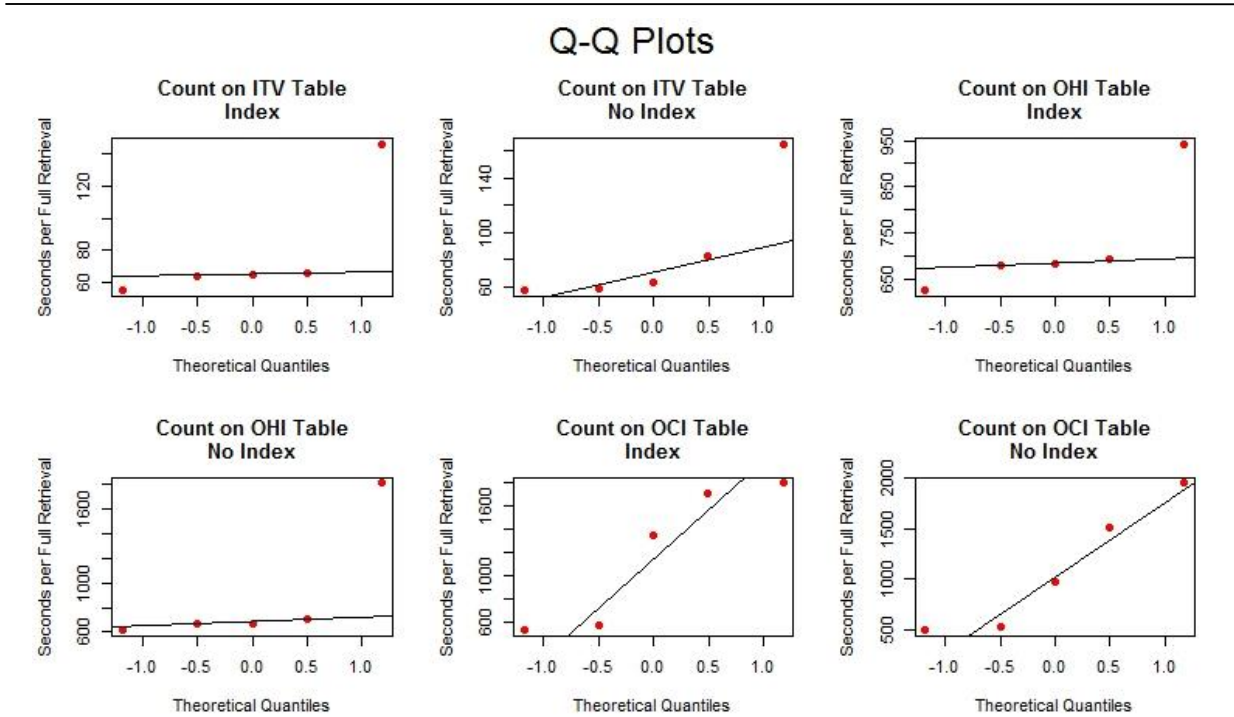
data: oci\$index and oci\$noIndex
t = 4.1641, df = 3, p-value = 0.0252
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval: 0.2213952 0.9950244

sample estimates: cor 0.9233116

Given a p-value critical level for rejection at .05, the null hypothesis indicating a poor relationship between indexed versus non-indexed features in our dataset was rejected. Furthermore, the values of the Pearson’s coefficients are 0.9825035, 0.9776403, and 0.9233116 respectively indicating *significant correlation between the tested variables*.

TABLE 2

Q-Q Plots



Given a perfect normal distribution in TABLE 2, the data points would fall exactly on the diagonal line. Many points are close, especially in the middle section, but the points in the tails are pretty far off. Too many points above the line would indicate a general skew to the left. And to reiterate, the null hypothesis (H_0) is that the average indexed field for billing, historical, and current items in our datasets makes record retrieval significantly faster than non-indexed record retrievals using 'select count (*)' versus 'select count *indexed field*' on the ITV_BASE, OHI_BASE, and OCI_BASE tables. Our alternative hypothesis (H_a) is that indexed field retrieval by way of select statements is NOT significantly different than non-indexed record retrievals. **Based on the similarities in speed of retrieval as evidenced by the Q-Q plots, our position is to reject the null hypothesis and accept the alternative hypothesis that selecting indexed fields does NOT make querying faster than counting all fields at once.**

Datasets

OHI Index	OHI No Index
943.35	1806.29
678.05	671.08
690.74	710.97
623.74	662.83
681.94	615.27
ITV Index	ITV No Index
145.72	164.99
66.14	82.75
55.37	57.24
64.54	58.29
65.59	63.62
OCI Index	OCI No Index
569.25	495.28

528.82	524.64
1354.08	966.15
1797.98	1515.85
1709.99	1946.50

Recommendations

My recommendation is to accept the alternative hypothesis that creating indexed fields does not make querying MCCDMPRD significantly faster and search for another solution to improve performance.