

Figure 1: Probability of a document being accessed on Day 8 as a function of the number of times it was accessed in the previous 7 days (for Frequencies < 100).

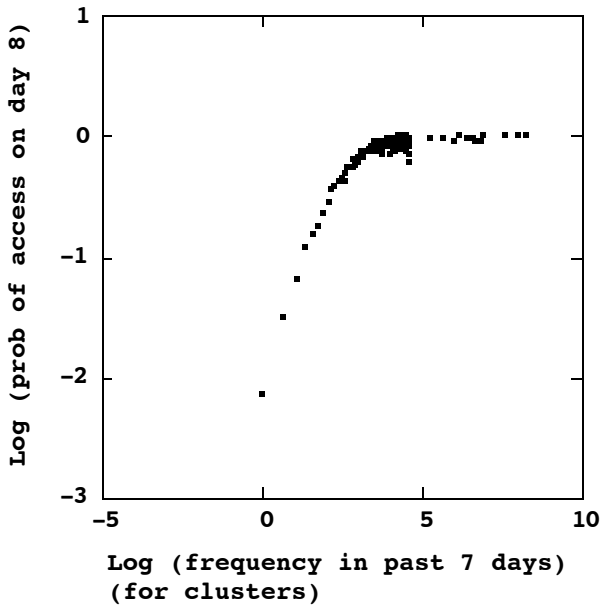


Figure 2: Log transform of probability of a document being accessed on Day 8 as a function of the number of times it was accessed in the previous 7 days (for clustered frequencies).

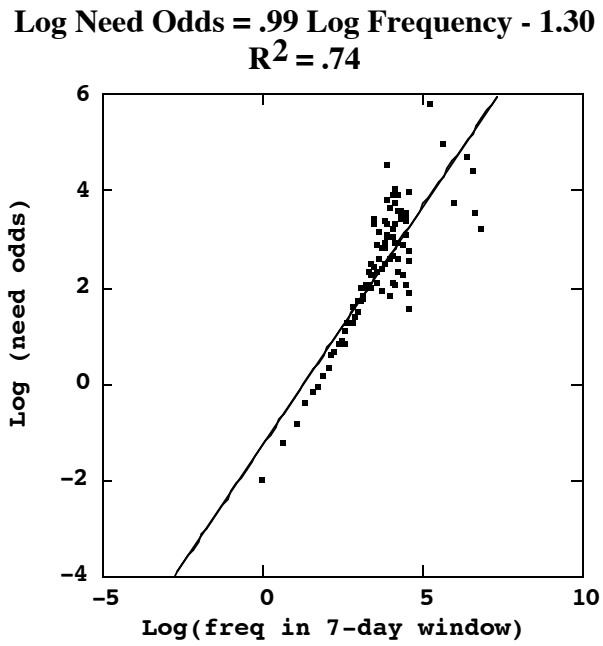


Figure 3: Transformation of plotting log need odds of a document access on Day 8 as a function of the log number of times it was accessed in the previous 7 days (for clustered frequencies).

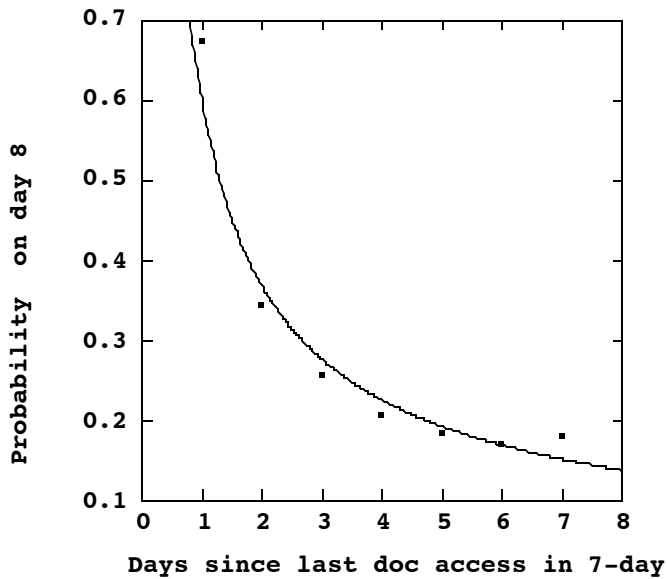


Figure 4: Probability of a document access on Day 8 as a function of how long it has been since the document was accessed in the previous 7 days.

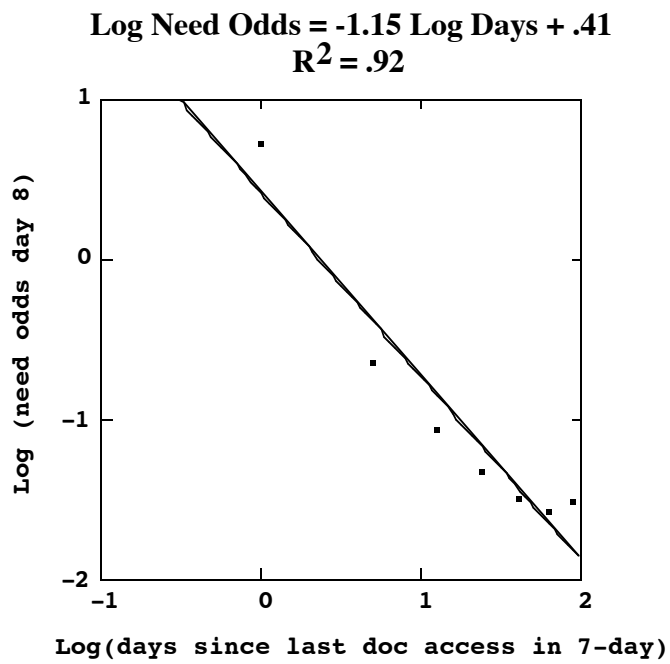


Figure 5: Transformation of plotting log need odds of a document access on Day 8 as a function of the log of how long it has been since the document was accessed in the previous 7 days.

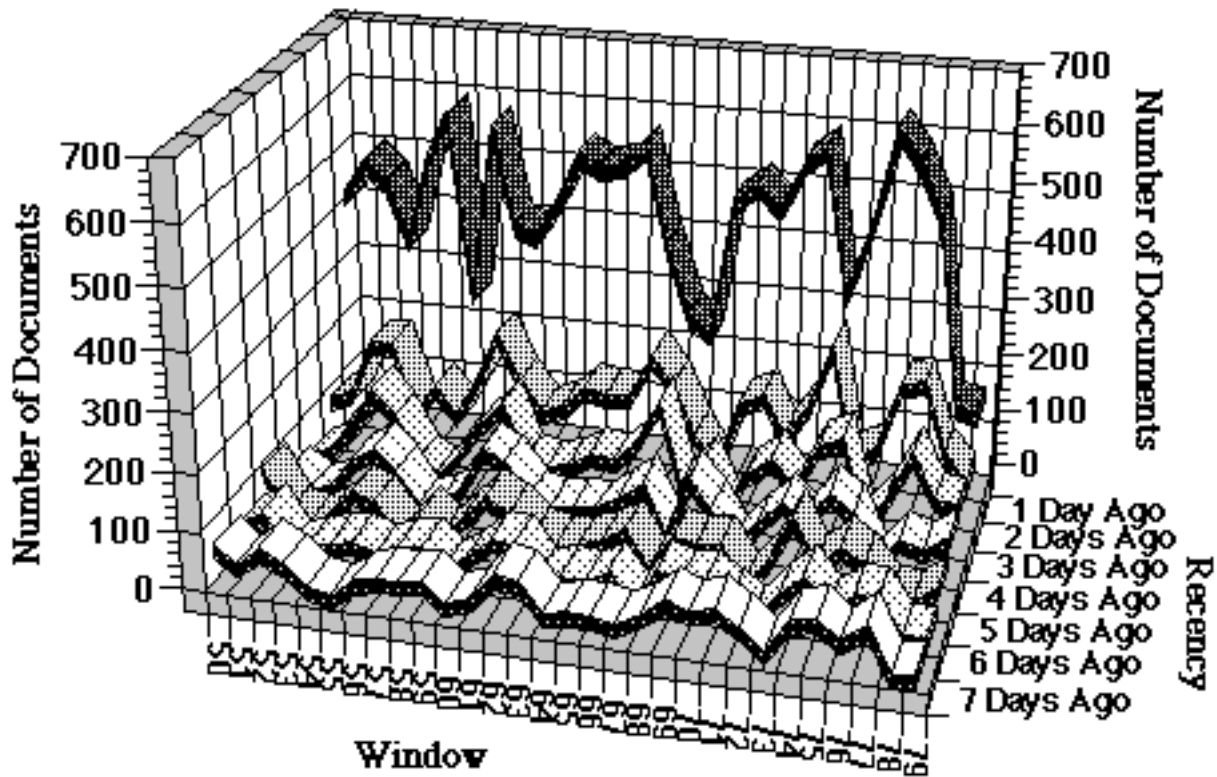


Figure 6: Plot of the number of documents for each day of recency across the last month of windows in the dataset. The graphing reveals a kind of cascading effect, where fluctuations in the number of documents requested one day ago are mirrored in the fluctuations for seven days ago.

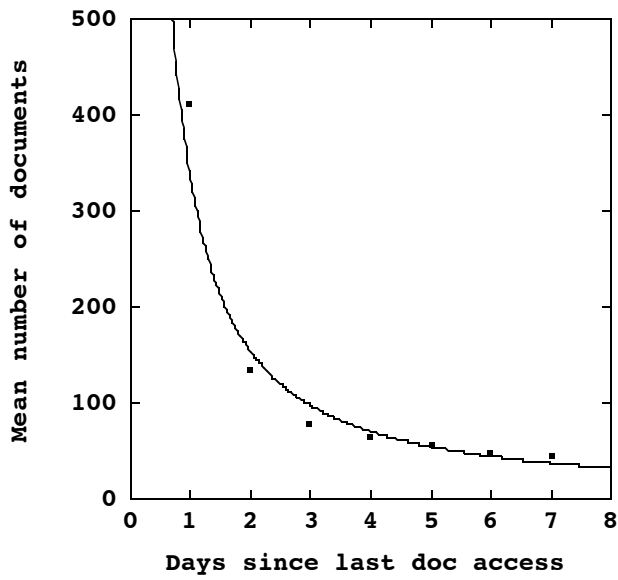


Figure 7: The mean number of documents for each day across windows as a function of how long it has been since the document was accessed in the previous 7 days.

$$\text{Log Mean Num} = 1.04 \text{ Log Days} - 5.65$$

$$R^2 = .99$$

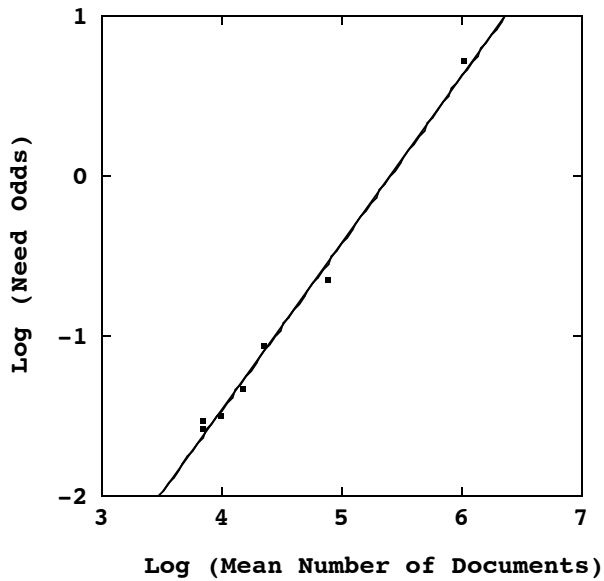


Figure 8: Transformation of plotting the log need odds as a function of the log of the mean number of documents for each day.

$$\text{Log Need Odds} = -.12 \text{ Log Mean Num} + 5.83$$

$$R^2 = .97$$

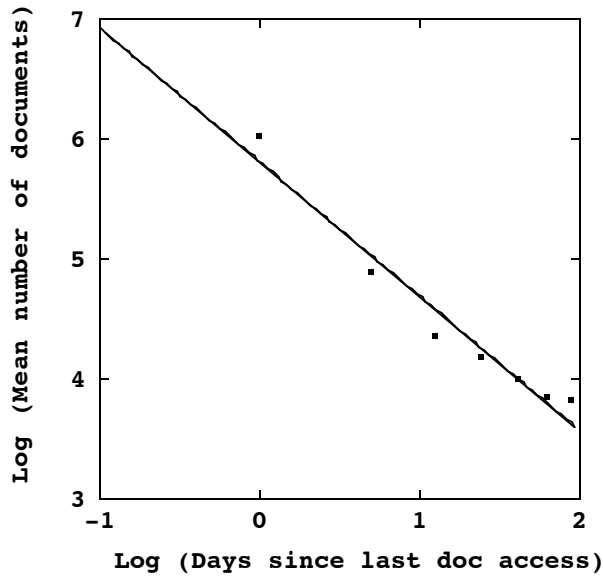


Figure 9: Transformation of plotting the log need odds as a function of the log of the mean number of documents for each day across windows.

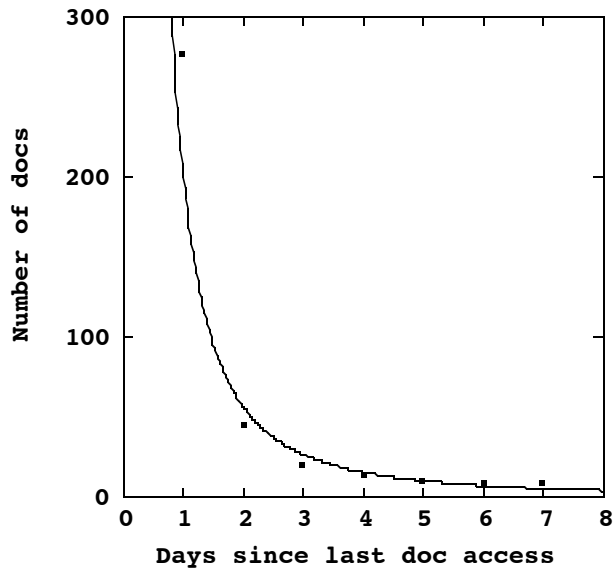


Figure 10: The number of documents that would correctly be cached for each day. Note that this plot essentially combines Figures 4 and Figure 7.

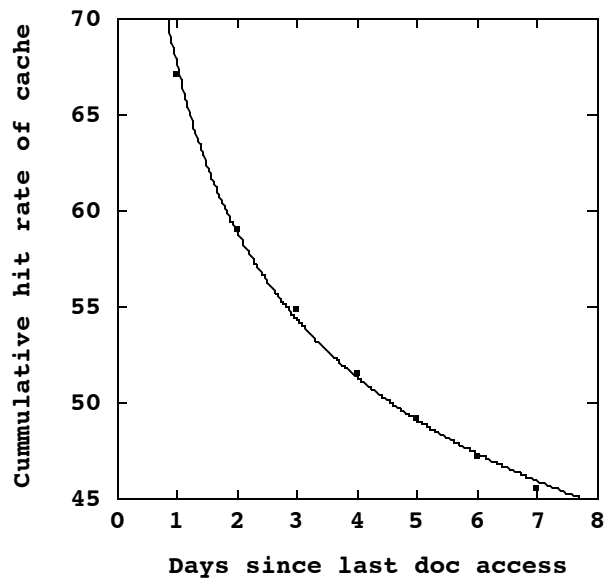


Figure 11: The hit rate of documents in the cache as a function of how long documents are kept in the cache (see text).

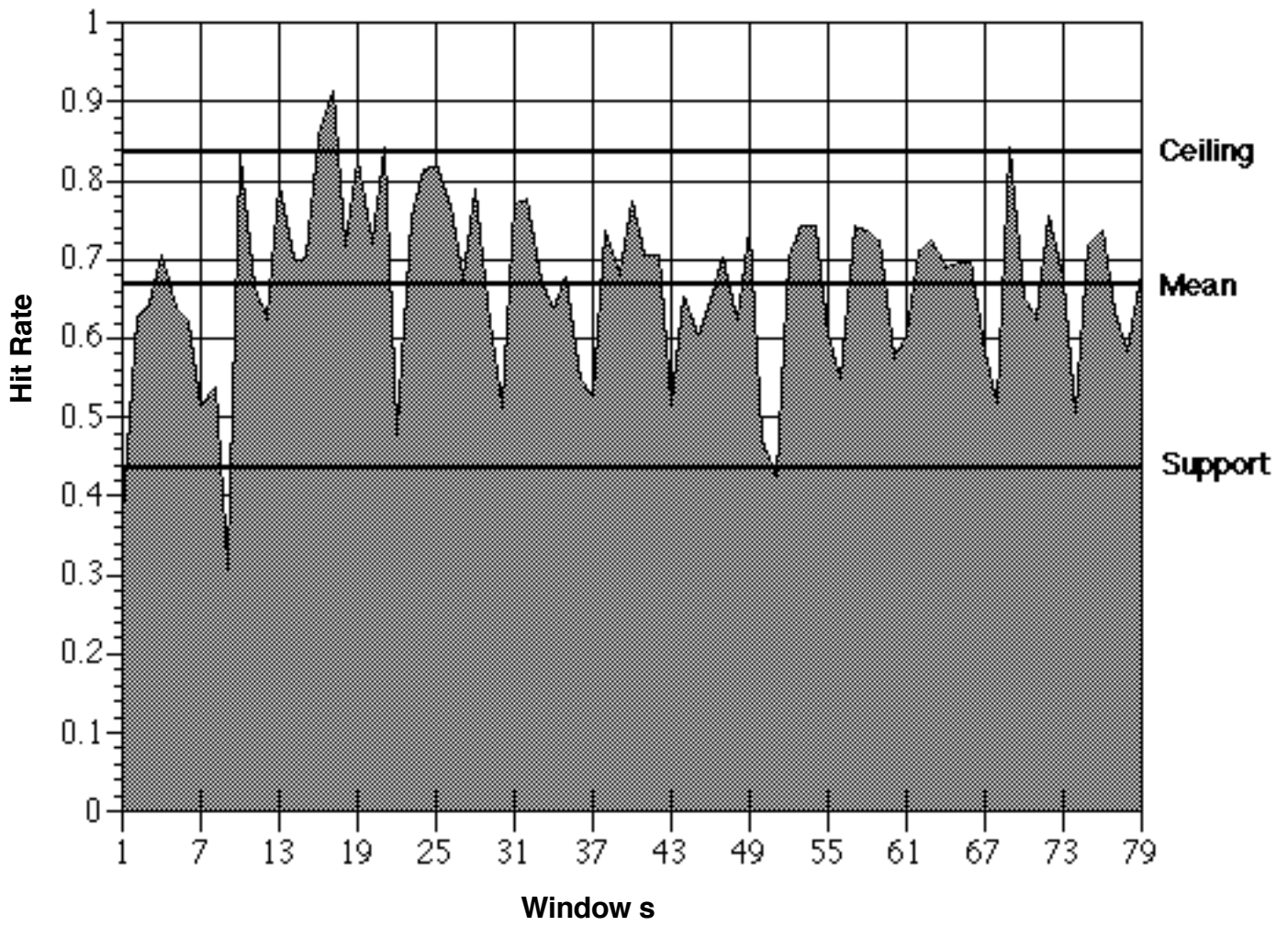


Figure 12: For all accesses with a recency of one day (i.e., the last day in the window) the Y-axis represents the hit rate, while the X-axis represents all windows in the dataset (see text).

```
if cache is full or end of day then
    day = window size

    while cache is above comfort level and day > 1 do
        remove files with recency == day
        day = day-1
    end while

    if day == 1 then
        while cache is above comfort level do
            remove largest files
        end if
    end if
end if
```

Figure 13. Sketch of the caching algorithm.