Finding Distressed Stocks

Aaron Hardy

2020-01-08

Finding Highly Risky Stocks

The Formula

Logit Probability of Financial Distress (LPFD)

A logistic regression model is used to estimate whether a firm is in financial distress or not.

$$\mathbf{LPFD} = -\mathbf{20.26} \times \mathbf{NIMTAAVG} + \mathbf{1.42} \times \mathbf{TLMTA} - \mathbf{7.13} \times \mathbf{EXRETAVG} + \mathbf{1.41} \times \mathbf{SIGMA} - \mathbf{0.045} \times \mathbf{RSIZE} - \mathbf{2.1}$$

The "LPFD" is then transformed into a probability of financial distress (PFD).

Probability of Financial Distress (PFD)

$$PFD = \frac{1}{1 + e^{-LPFD}}$$

Weighting the value

$$XAVG = .5333 \times t + .2666 \times (t-1) + .1333 \times (t-2) + .0666 \times (t-3)$$

The Data

connect to a database and load some data

```
# Clear Workspace
rm(list=ls(all=TRUE))


# Load libraries ----------------------------------------------------------
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(RMySQL)
```

```
## Loading required package: DBI
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```
library(tidyr)

# Establish a MySQL connection
con <- dbConnect(MySQL(), dbname = "financial_data", user = 'root' , password='paperback12')

# dbListTables(con)

# Create a function to pull multiple fields from SQL tables
get_data <- function(conn, field) {
      tbl(conn, field) %>% as_tibble() %>% mutate(date = ymd(date))
}

fields <- c("market_cap", "enterprise_value", "total_liabilities", "cash")

list_data <- list() # Initialize the container
for(i in seq_along(fields)) {
      table_data <- get_data(con, fields[i])
      list_data <- c(list_data, list(table_data))
}

all_data <- Reduce(function(tb1, tb2) full_join(tb1, tb2, by = c("id", "date")), list_data)
cleaned_data <- all_data %>% na.omit() %>% group_by(id)

# dbListTables(con)


dbDisconnect(con) # Disconnect from SQL database
```

```
## [1] TRUE
```