

# Programming with Data

## Session 8: Bankruptcy Predictions

**Dr. Wang Jiwei**

**Master of Professional Accounting**

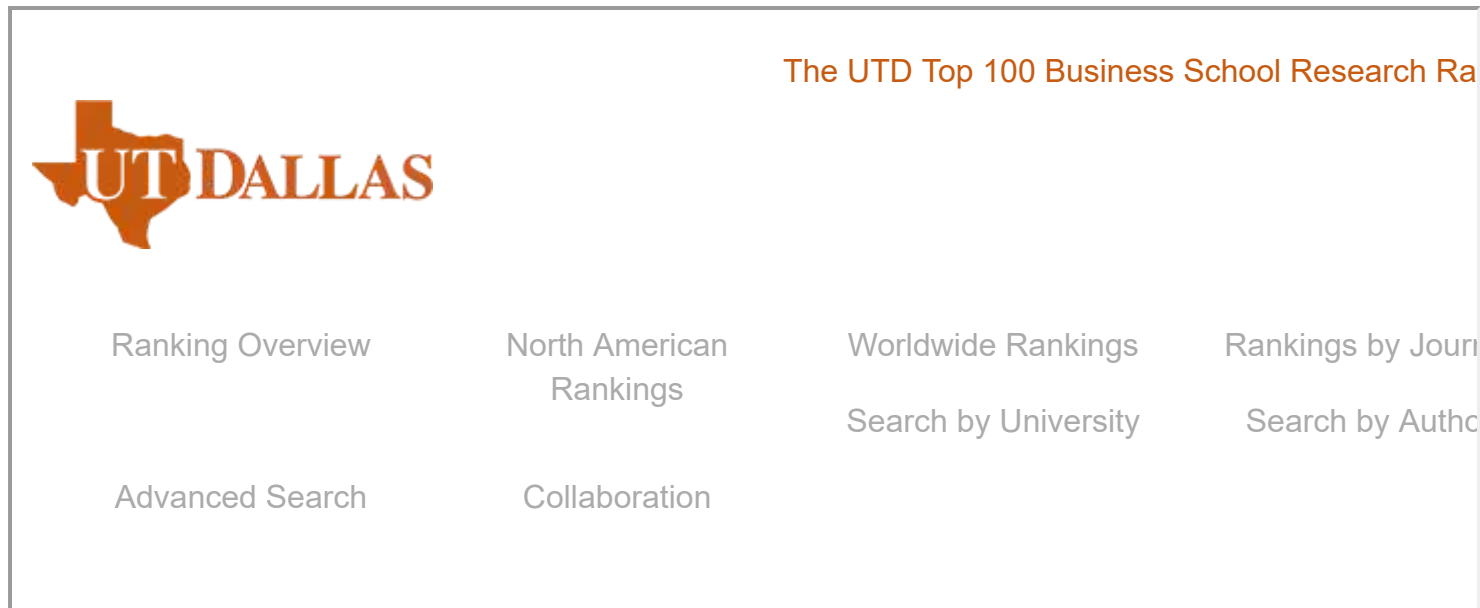


# **Academic research in business**


# Academic research in business

- Analytical/Theory
  - Pure economics/mathematics proofs and simulation
- Experimental
  - Proper experimentation done on individuals or groups
- Archival/Empirical
  - Data driven and requires data analytics skills

■ SMU ranked 2nd in Asia in business research



The UTD Top 100 Business School Research Ra



Ranking Overview      North American Rankings      Worldwide Rankings      Rankings by Journal

Advanced Search      Collaboration      Search by University      Search by Author

# Academic research in accounting

- Accounting research builds on work from many fields:
  - Economics
  - Finance
  - Psychology
  - Econometrics/Statistics/Mathematics
  - Computer science (more recently)

■ SMU ranked 2nd worldwide in archival accounting research

## Main Accounting Rankings for Universities: 2020 (</rankings/univrank/updates.php>)

Go to Citation-Based Rankings ([/rankings/univrank/rankings\\_ct.php](/rankings/univrank/rankings_ct.php))

The rankings presented via the links below are based on the award winning research (</rankings/univrank/authorbios.php>). These rankings are based on classifications of peer reviewed articles in 12 accounting journals since 1990. To see the set of rankings that are of interest to you, click on the appropriate title. Learn more about this website. (<https://youtu.be/-8Wfv-4TdVM>)

All	AIS	Audit	Financial	Managerial	Tax	Other Topic
Analytical	AIS	Audit	Financial	Managerial	Tax	Other

# Where to find academic research

- The **SMU library** has access to seemingly all high quality business research
  - **50 Business Journals used in FT Research Rank**
- **Google Scholar** is a great site to discover research past and present
- **SSRN** is the site to find cutting edge research in business and social sciences
  - **List of top accounting papers on SSRN** (by downloads)
- Research helps to find good predictors and build better models
  - "literature review" in academic jargon
  - aka "Standing on the shoulders of giants"
  - very helpful to build your mental models:
    - what drive new information disclosure?
    - financial statement restatement?
    - social media following?

# **Academic models for bankruptcy: Altman Z-Score**

# Where does the model come from?

- Altman 1968, Journal of Finance
- A seminal paper in Accounting and Finance cited over 15,000 times by other academic papers
- The model was developed to identify firms likely to go bankrupt from a pool of firms
- Focuses on using financial ratio analysis to determine such firms
- Click the image to read the paper

## *The Journal of* FINANCE

Vol. XXIII

SEPTEMBER 1968

No. 4

### FINANCIAL RATIOS, DISCRIMINANT ANALYSIS AND THE PREDICTION OF CORPORATE BANKRUPTCY

EDWARD I. ALTMAN\*

ACADEMICIANS SEEM to be moving toward the elimination of ratio analysis as an analytical technique in assessing the performance of the business enterprise. Theorists downgrade arbitrary rules of thumb, such as company ratio comparisons, widely used by practitioners. Since attacks on the relevance of ratio analysis emanate from many esteemed members of the scholarly world, does this mean that ratio analysis is limited to the world of "nuts and bolts"? Or, has the significance of such an approach been unattractively garbed and therefore unfairly handicapped? Can we bridge the gap, rather than sever the link, between traditional ratio "analysis" and the more rigorous statistical techniques which have become popular among academicians in recent years?

The purpose of this paper is to attempt an assessment of this issue—the quality of ratio analysis as an analytical technique. The prediction of corporate bankruptcy is used as an illustrative case.<sup>1</sup> Specifically, a set of financial and economic ratios will be investigated in a bankruptcy prediction context wherein a multiple discriminant statistical methodology is employed. The data used in the study are limited to manufacturing corporations.

A brief review of the development of traditional ratio analysis as a technique for investigating corporate performance is presented in section I. In section II the shortcomings of this approach are discussed and multiple discriminant analysis is introduced with the emphasis centering on its compatibility with ratio analysis in a bankruptcy prediction context. The discriminant model is developed in section III, where an initial sample of sixty-six firms is utilized to establish a function which best discriminates between companies in two mutually exclusive groups: bankrupt and non-bankrupt firms. Section IV reviews empirical results obtained from the initial sample and several secondary samples, the latter being selected to examine the reliability of the discriminant

\* Assistant Professor of Finance, New York University. The author acknowledges the helpful suggestions and comments of Keith V. Smith, Edward F. Renshaw, Lawrence S. Ritter and the *Journal's* reviewer. The research was conducted while under a Regents Fellowship at the University of California, Los Angeles.

1. In this study the term bankruptcy will, except where otherwise noted, refer to those firms that are legally bankrupt and either placed in receivership or have been granted the right to reorganize under the provisions of the National Bankruptcy Act.

# Model specification

$$Z = 1.2x_1 + 1.4x_2 + 3.3x_3 + 0.6x_4 + 0.999x_5$$

- $x_1$ : Working capital/Total assets
- $x_2$ : Retained earnings/Total assets
- $x_3$ : Earnings before interest and taxes/Total assets
- $x_4$ : Market value equity/Book value of total debt
- $x_5$ : Sales/Total assets

■ This looks like a linear regression without a constant



# How did the measure come to be?

- It actually isn't a linear regression
    - It is a clustering method called MDA (Multiple Discriminant Analysis)
      - There are newer methods these days, such as SVM (Support Vector Machine)
  - Used data from 1946 through 1965
    - 33 US manufacturing firms that went bankrupt, 33 that survived
  - More about this, from Altman himself in 2000: [Altman 2000](#)
    - Read the section "Variable Selection" starting on page 8
      - Skim through  $x_1$ ,  $x_2$ ,  $x_3$ ,  $x_4$ , and  $x_5$  if you are interested in the ratios
- How would these assumptions stand today?

# How to use it?

---

Altman Z Score	Meaning of the cut-off points
----------------	-------------------------------

---

$Z > 2.67$	Non-distress Zones
------------	--------------------

$1.81 < Z < 2.67$	Grey Zones
-------------------	------------

$Z < 1.81$	Distress Zones
------------	----------------

---

# Who uses it?

- Despite the model's simplicity and age, it is still in use
  - The simplicity of it plays a large part
- Frequently used by financial analysts, especially credit analysts

■ Recent news mentioning it



# Application

# Main question

| Can we use bankruptcy models to predict supplier bankruptcies?

But first:

| Does the Altman Z-score [still] pick up bankruptcy?

| Is this a forecasting or forensics question?

- It has a time dimension like a forecasting question
- It has a feeling of a forensics question

# The data

- Compustat provides data on bankruptcies, including the date a company went bankrupt
  - Bankruptcy information is included in the "footnote" items in Compustat
    - If `dlsrn == 2`, then the firm went bankrupt
    - Bankruptcy date is `dldte`
- All components of the Altman Z-Score model are in Compustat
  - But we'll pull market value from CRSP, since it is more complete
- All components of our later models are from Compustat as well
- Company credit rating data also from Compustat (Rankings)

# Bankruptcy Law

- In the U.S.A.
  - Chapter 7 of the Bankruptcy Code
    - The company ceases operating and liquidates
  - Chapter 11 of the Bankruptcy Code
    - For firms intending to reorganize the company to "try to become profitable again" (US SEC)
- In Singapore
  - PART X of the Companies Act (Cap. 50)
  - What are the stages involved in a liquidation?



# Common outcomes of bankruptcy

1. Cease operations entirely (liquidated)
  - In which case the assets are often sold off
2. Acquired by another company
3. Merge with another company
4. Successfully restructure and continue operating as the same firm
5. Restructure and operate as a new firm





# Calculating bankruptcy

```
# initial cleaning
df <- df %>% filter(at >= 1, revt >= 1)

## Merge in stock value
df$date <- as.Date(df$datadate)
df_mve$date <- as.Date(df_mve$datadate)
df_mve <- df_mve %>% rename(gvkey = GVKEY) # df_mve uses GVKEY, df uses gvkey
df_mve$MVE <- df_mve$csho * df_mve$prcc_f # MVE = no. of shares * price per share

df <- left_join(df, df_mve[, c("gvkey", "date", "MVE")])

df <- df %>%
  group_by(gvkey) %>%
  mutate(bankrupt = ifelse(row_number() == n() & dlrns == 2 &
    !is.na(dlrns), 1, 0)) %>%
  ungroup() #set the most recent year as the bankruptcy year
prop.table(table(df$bankrupt)) # proportion in a table format
```

```
##
##           0           1
## 0.997779917 0.002220083
```

- `row_number()` gives the current row within the group, with the first row as 1, next as 2, etc.
- `n()` gives the number of rows in the group

# Calculating the Altman Z-Score

```
df <- df %>% # Calculate the measures needed
  mutate(wcap_at = wcap / at, # x1
         re_at = re / at, # x2
         ebit_at = ebit / at, # x3
         mve_lt = MVE / lt, # x4
         revt_at = revt / at) # x5

# cleanup
df <- df %>% # to replace all infinite numbers with NA
  mutate_if(is.numeric, list(~replace(., !is.finite(.), NA)))

# Calculate the score
df <- df %>%
  mutate(Z = 1.2 * wcap_at + 1.4 * re_at + 3.3 * ebit_at +
         0.6 * mve_lt + 0.999 * revt_at)

# Calculate date info for merging
df$date <- as.Date(df$datadate)
df$year <- year(df$date)
df$month <- month(df$date)
```

- Calculate  $x_1$  through  $x_5$
- Apply the model directly

# Build in credit ratings

| We'll check our Z-score against credit rating as a simple validation

```
# df_ratings has credit ratings from Compustat

# Ratings, in order from worst to best
ratings <- c("D", "C", "CC", "CCC-", "CCC", "CCC+", "B-", "B", "B+", "BB-",
            "BB", "BB+", "BBB-", "BBB", "BBB+", "A-", "A", "A+", "AA-", "AA",
            "AA+", "AAA-", "AAA", "AAA+")

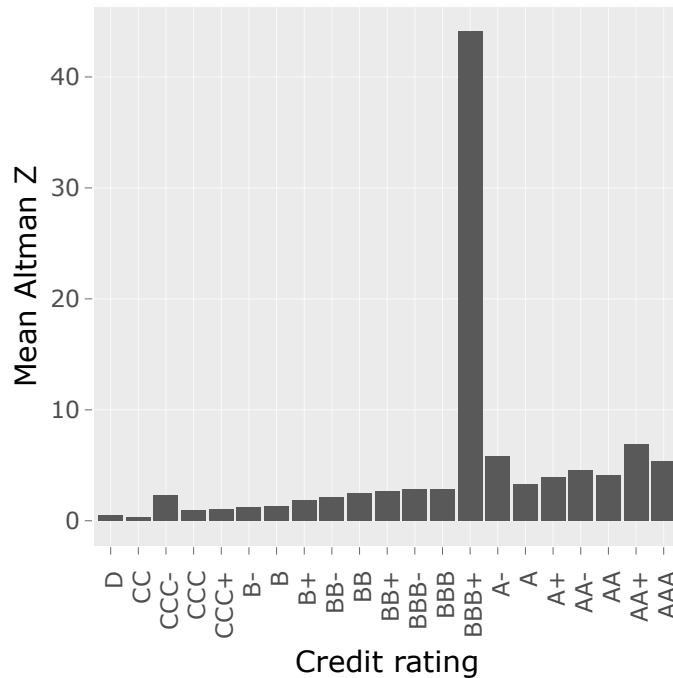
# Convert string ratings (splticrm) to ordered factor ratings
df_ratings$rating <- factor(df_ratings$splticrm, levels = ratings, ordered = T)

df_ratings$date <- as.Date(df_ratings$datadate)
df_ratings$year <- year(df_ratings$date)
df_ratings$month <- month(df_ratings$date)

# Merge together data
df <- left_join(df, df_ratings[ , c("gvkey", "year", "month", "rating")])

## Joining, by = c("gvkey", "year", "month")
```

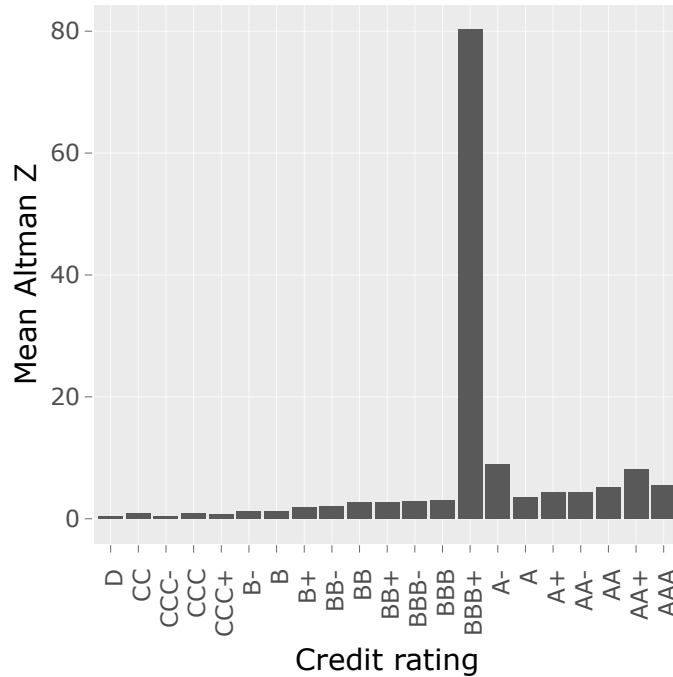
# Z vs credit ratings, 1973-2017



```
df %>%  
  filter(!is.na(Z),  
         !is.na(bankrupt)) %>%  
  group_by(bankrupt) %>%  
  mutate(mean_Z=mean(Z,na.rm=T)) %>%  
  slice(1) %>%  
  ungroup() %>%  
  select(bankrupt, mean_Z) %>%  
  html_df()
```

bankrupt	mean_Z
0	4.424812
1	0.927843

# Z vs credit ratings, 2000-2017



```
df %>%  
  filter(!is.na(Z),  
         !is.na(bankrupt),  
         year >= 2000) %>%  
  group_by(bankrupt) %>%  
  mutate(mean_Z=mean(Z,na.rm=T)) %>%  
  slice(1) %>%  
  ungroup() %>%  
  select(bankrupt, mean_Z) %>%  
  html_df()
```

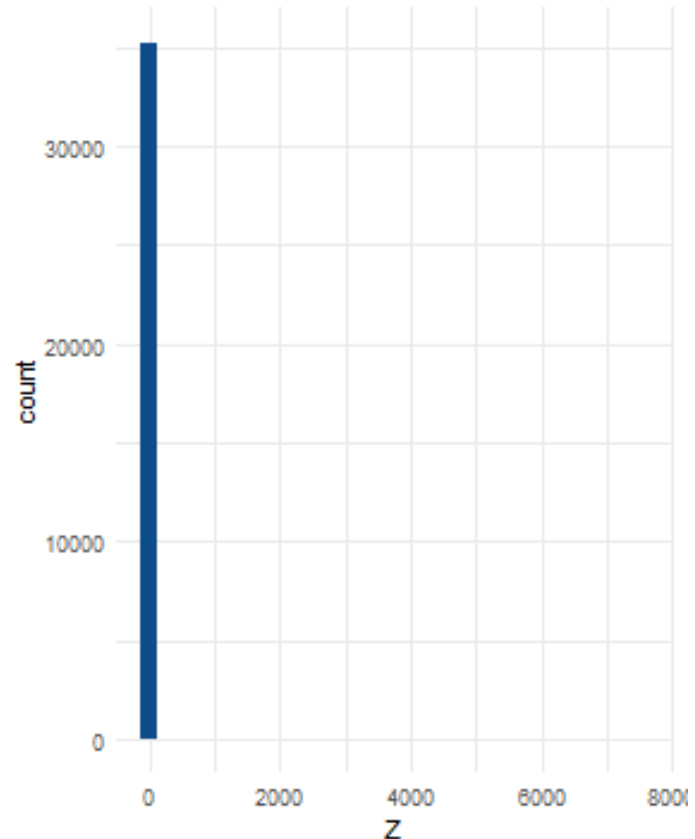
bankrupt	mean_Z
0	5.346655
1	1.417683

# Outlier detection with descriptive statistics

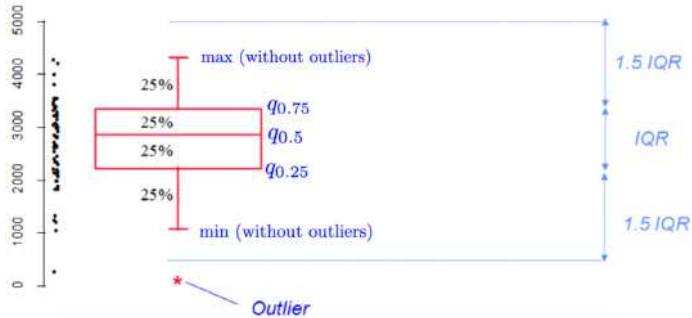
## ■ Summary Statistics

```
##          Z
## Min.    :-116.095
## 1st Qu.:  2.114
## Median :  3.192
## Mean    :  4.417
## 3rd Qu.:  4.648
## Max.    :7390.498
## NA's    :15591
```

## ■ Histogram



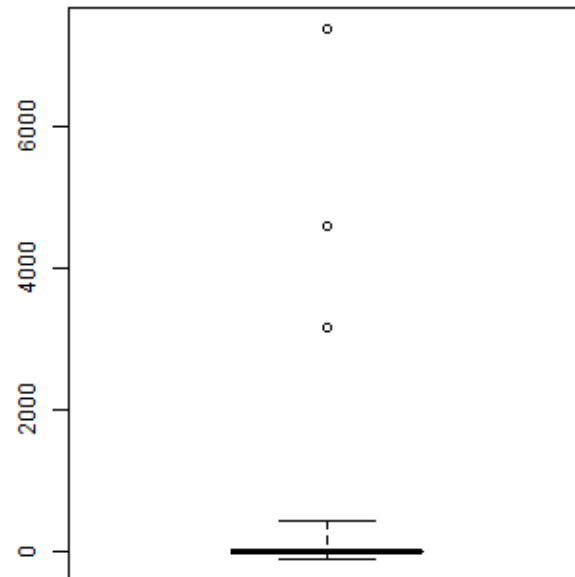
# Outlier detection with boxplot



```
boxplot(df$Z, ylab = "Z",  
        col = "red", range = 171,  
        main = "Boxplot of Z Score")
```

- Interquartile range (IQR) criterion
- IQR is the difference between the third and first quartile  $q_{0.75} - q_{0.25}$
- Outlier: outside  
 $I = [q_{0.25} - 1.5 \cdot IQR; q_{0.75} + 1.5 \cdot IQR]$
- Observations considered as potential outliers by the IQR criterion are displayed as points in the boxplot.

Boxplot of Z Score



# Output from boxplot

```
boxplot.stats(df$Z, coef = 171) # default coef = 1.5
```

```
## $stats  
## [1] -116.094972    2.113688    3.192143    4.648596  437.655080  
##  
## $n  
## [1] 35308  
##  
## $conf  
## [1] 3.170828 3.213457  
##  
## $out  
## [1] 4604.166 3148.402 7390.498
```

- **stats**: the extreme of the lower whisker (min), the lower hinge ( $q_{0.25}$ ), the median, the upper hinge ( $q_{0.75}$ ) and the extreme of the upper whisker (max).
- **n**: the number of non-NA observations in the sample.
- **conf**: the lower and upper extremes of the notch (expected range of variability of the median, plotted by narrowing the box around the median, for visual comparison of medians among multiple boxes with 95% confidence)
- **out**: the values of any data points which lie beyond the extremes of the whiskers



# Output outliers

```
out <- boxplot.stats(df$Z, coef = 171)$out
# Return the index of outliers
out_ind <- which(df$Z %in% c(out))
html_df(df[out_ind, c("gvkey", "conm", "year", "Z", "bankrupt", "rating")])
```

gvkey	conm	year	Z	bankrupt	rating
100338	RELX PLC	2012	4604.166	0	BBB+
100338	RELX PLC	2013	3148.402	0	BBB+
100338	RELX PLC	2014	7390.498	0	BBB+

- More brutal way of treating outliers based on descriptive statistics
  - **truncation/trimming**: remove top and bottom 1% of observations
  - **winsorizing**: replace top and bottom 1% values with the 99th and 1st percentile values

# Outliers detection with stat tests

- No formal definition of outliers, hence no rigorous statistical tests
- Some stat tests by assuming normal distribution, [read here](#)
  - **Rosner's Test**

```
library(EnvStats)
```

```
# k is the number of suspected outliers, default k = 3
```

```
test <- rosnerTest(df$Z, k = 3)
```

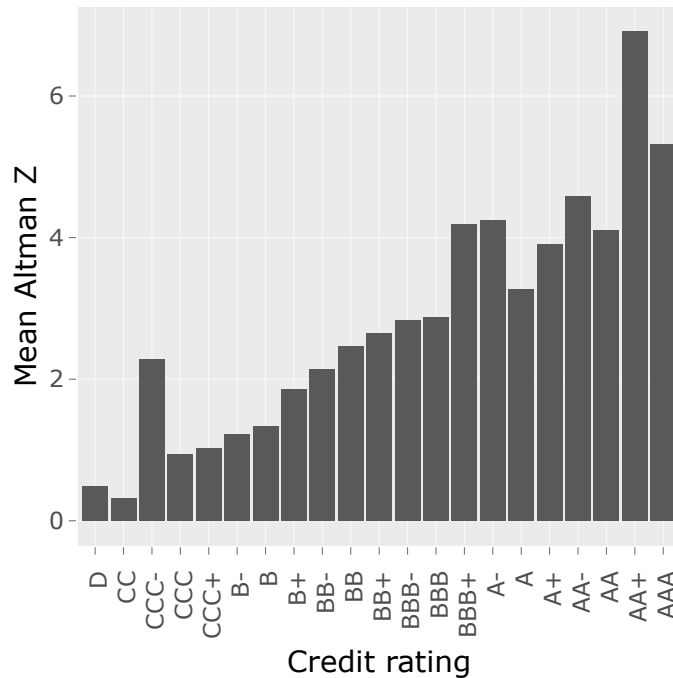
```
test$all.stats
```

```
##   i   Mean.i     SD.i   Value Obs.Num   R.i+1 lambda.i+1 Outlier
## 1 0 4.417285 49.81383 7390.498  45975 148.2737  4.821964    TRUE
## 2 1 4.208089 30.59841 4604.166  45973 150.3332  4.821958    TRUE
## 3 2 4.077801 18.35578 3148.402  45974 171.2988  4.821952    TRUE
```

- Treatment of outliers is subjective. In general, supervised regression models are more sensitive to outliers, unsupervised classification algos are more robust to outliers

**|** We deleted gvkey == 100338

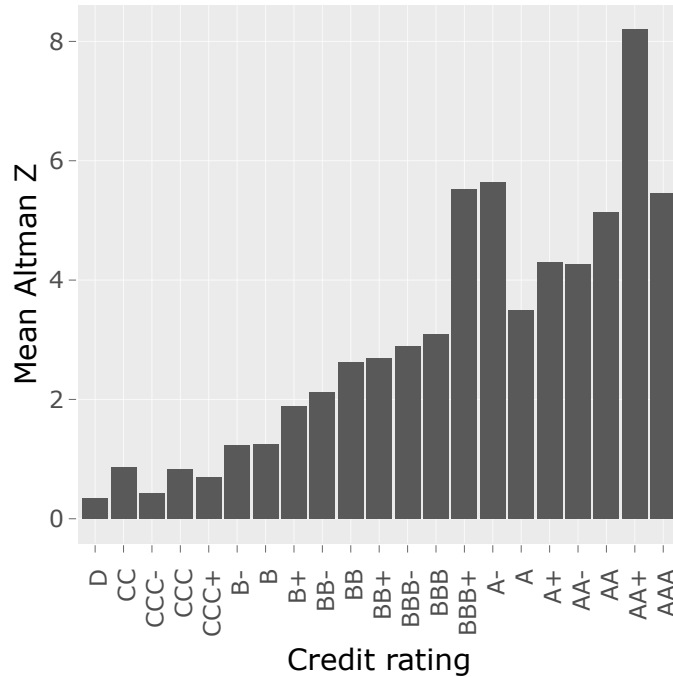
# Z vs credit ratings, 1973-2017



```
df %>%  
  filter(!is.na(Z),  
         !is.na(bankrupt)) %>%  
  group_by(bankrupt) %>%  
  mutate(mean_Z=mean(Z,na.rm=T)) %>%  
  slice(1) %>%  
  ungroup() %>%  
  select(bankrupt, mean_Z) %>%  
  html_df()
```

bankrupt	mean_Z
0	3.939223
1	0.927843

# Z vs credit ratings, 2000-2017



```
df %>%  
  filter(!is.na(Z),  
         !is.na(bankrupt),  
         year >= 2000) %>%  
  group_by(bankrupt) %>%  
  mutate(mean_Z=mean(Z,na.rm=T)) %>%  
  slice(1) %>%  
  ungroup() %>%  
  select(bankrupt, mean_Z) %>%  
  html_df()
```

bankrupt	mean_Z
0	3.822281
1	1.417683

# Test it with a regression

```
fit_Z <- glm(bankrupt ~ Z, data = df, family = binomial)
summary(fit_Z)
```

```
##
## Call:
## glm(formula = bankrupt ~ Z, family = binomial, data = df)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.8297  -0.0676  -0.0654  -0.0624   3.7794
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.94354    0.11829 -50.245  < 2e-16 ***
## Z           -0.06383    0.01239  -5.151 2.59e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1085.2  on 35296  degrees of freedom
## Residual deviance: 1066.5  on 35295  degrees of freedom
## (15577 observations deleted due to missingness)
## AIC: 1070.5
##
## Number of Fisher Scoring iterations: 9
```

# So what?

- Read this article
  - "Carillion's liquidation reveals the dangers of shared sourcing"

Based on this article, why do we care about bankruptcy risk for other firms?

# Errors in binary testing

# Types of errors

		Prediction	
		Classify as success (i.e., positive)	Classify as failure (i.e., negative)
Actual observation	Actually a success	Correct (True Positive)	Type II error (False Negative)
	Actually a failure	Type I error (False Positive)	Correct (True Negative)

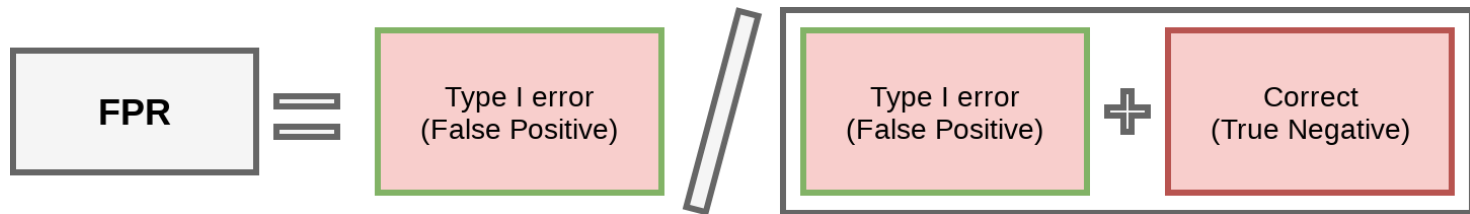
This is called a *Confusion Matrix*



# Type I error (False positive)

| We say that the company will go bankrupt, but they don't

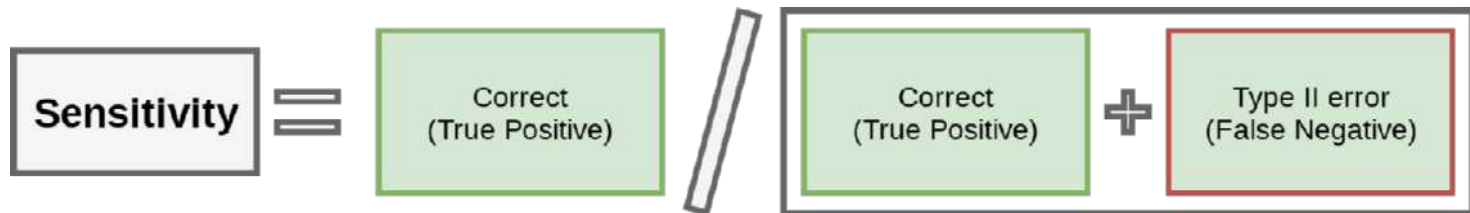
- A Type I error occurs any time we say something is *true*, yet it is false
- Quantifying type I errors in the data
  - False positive rate (FPR)
    - The percent of failures misclassified as successes
  - Specificity:  $1 - FPR$ 
    - A.k.a. true negative rate (TNR)
    - The percent of failures properly classified



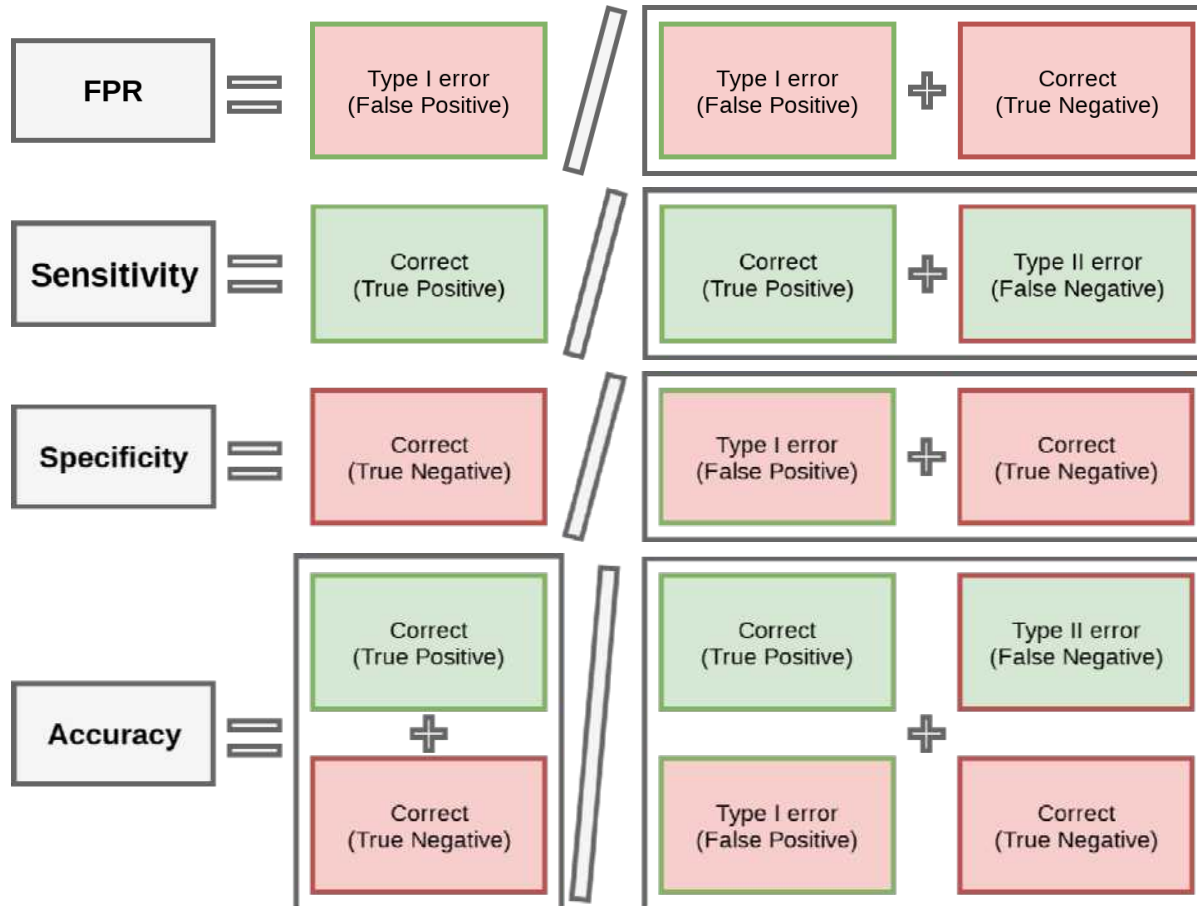
# Type 2 error (False negative)

| We say that the company *will not* go bankrupt, yet they do

- A Type II error occurs any time we say something is *false*, yet it is true
- Quantifying type I errors in the data
  - False negative rate (FNR):  $1 - \text{Sensitivity}$ 
    - The percent of successes misclassified as failures
  - Sensitivity:
    - A.k.a. true positive rate (TPR)
    - The percent of successes properly classified



# Useful equations



# A note on the equations

- Accuracy is very useful if you are predicting something that occurs reasonably frequently
  - Not too often, but not too rarely, say, at least 10% positive
  - e.g., a rare event of 1% positive, if we simply predict every single observation as a negative instance, you will get  $TP = 0$  and  $TN = 99$ , with accuracy of 99%.
- Sensitivity is very useful for rare events (TP is more important)
- Specificity is very useful for frequent events (TN is more important)
  - Or for events where misclassifying the null is relatively very costly
    - Criminal trials
    - Medical diagnoses

# Let's plot TPR and FPR out

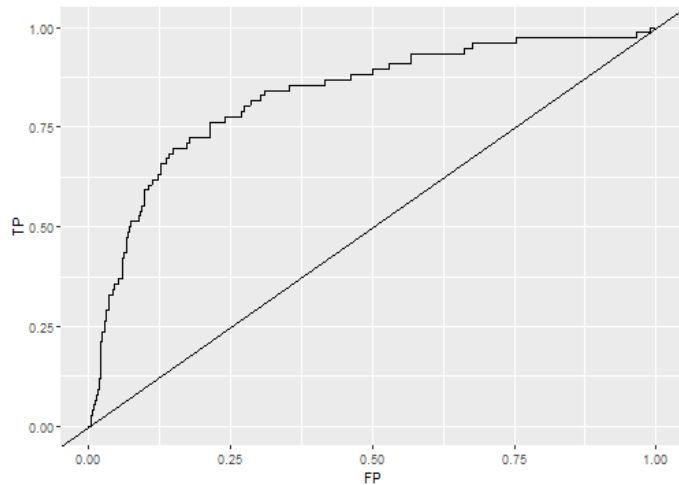
```
# ROCR 1.0-11 requires manual removal of NA in the prediction() function
# Suggest to install the 1.0-7 version from the archive
# https://cran.r-project.org/src/contrib/Archive/ROCR/ROCR_1.0-7.tar.gz
library(ROCR)
dfZ <- df %>% filter(!is.na(bankrupt), !is.na(Z))
pred_Z <- predict(fit_Z, dfZ, type = "response")
ROCpred_Z <- prediction(as.numeric(pred_Z), as.numeric(dfZ$bankrupt))
ROCperf_Z <- performance(ROCpred_Z, 'tpr', 'fpr')
```

- **package:ROCR** can calculate these for us!
  - **Other packages:** **package:pROC**, **PRROC**, and others.
- Notes on **package:ROCR**:
  1. The functions are rather picky and fragile
    - The vectors passed to **prediction()** aren't explicitly numeric
    - There are NAs in the data
  2. **prediction()** does not actually predict -- it builds an object based on your prediction (first argument) and the actual outcomes (second argument)
  3. **performance()** has more than 30 measures
    - 'tpr' is true positive rate
    - 'fpr' is false positive rate

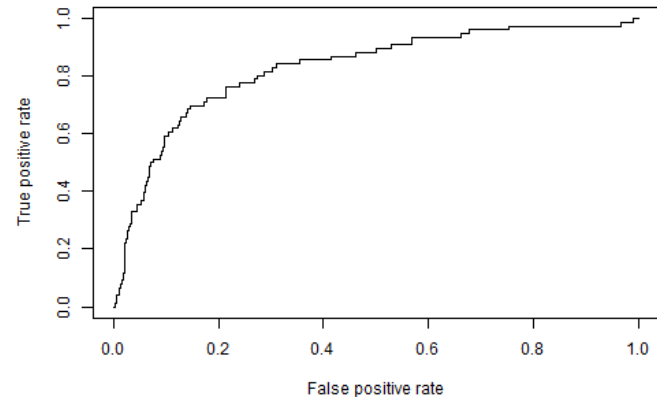
# Let's plot TPR and FPR out

- Two ways to plot it out:

```
df_ROC_Z <- data.frame(  
  FP = c(ROCperf_Z@x.values[[1]]),  
  TP = c(ROCperf_Z@y.values[[1]]))  
ggplot(data = df_ROC_Z,  
  aes(x = FP, y = TP)) +  
  geom_line() +  
  geom_abline(slope = 1)
```

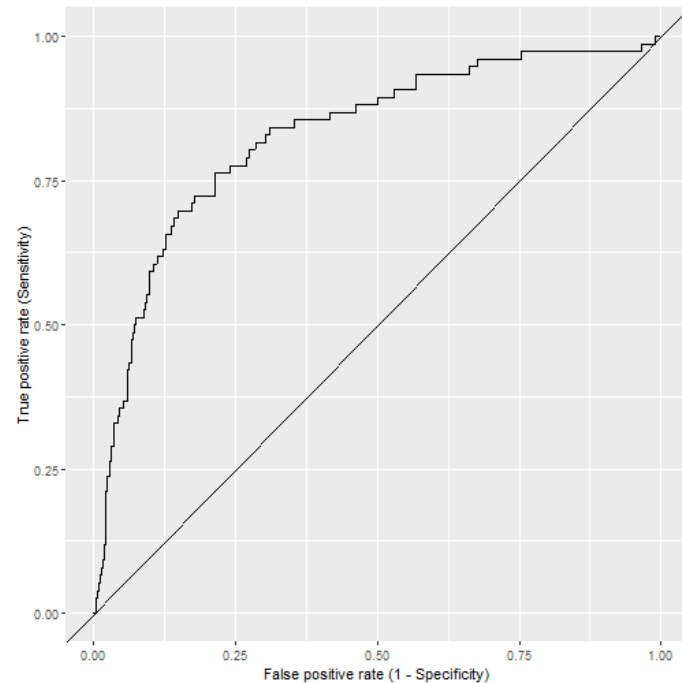


```
plot(ROCperf_Z)
```



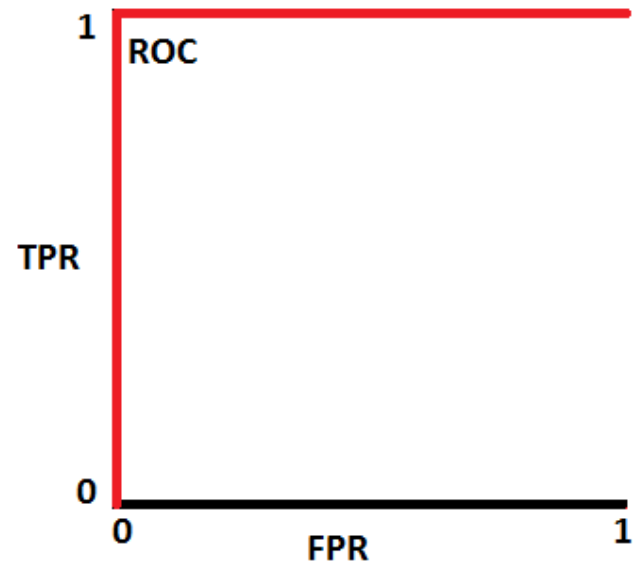
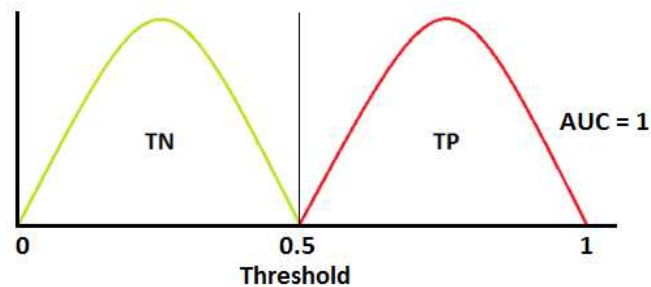
# ROC curves

- The previous graph is called a ROC curve, or *Receiver Operator Characteristic curve*
- The higher up and left the curve is, the better the model fits.
- Neat properties:
  - The area under a perfect model is always 1
  - The area under random chance is always 0.5
- **An Introduction to ROC Analysis**



# ROC curves when perfectly correct

- Red distribution curve is of the positive class
- Green distribution curve is of negative class
- If the model is perfect, ie, 100% correctly separate positive from negative

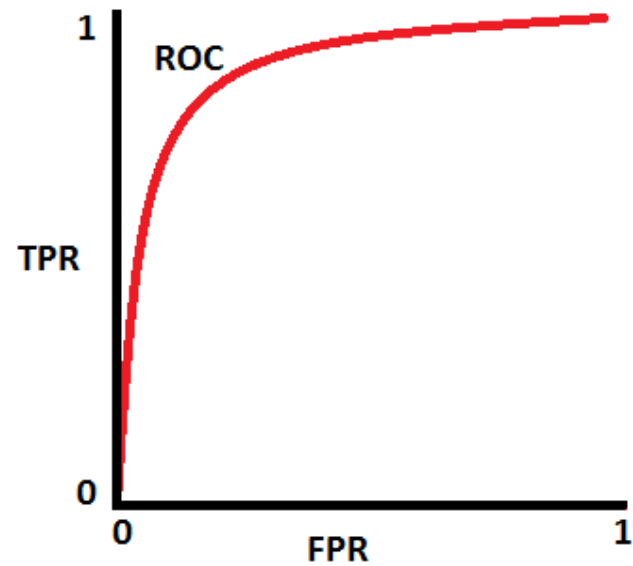
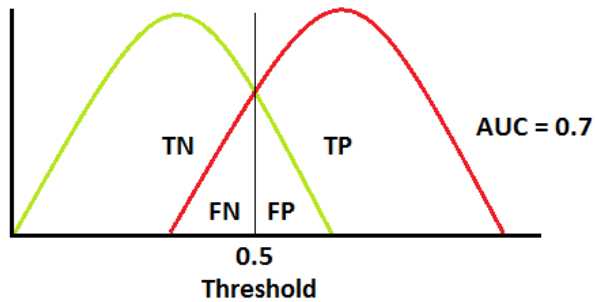


- The figures are from [here](#)



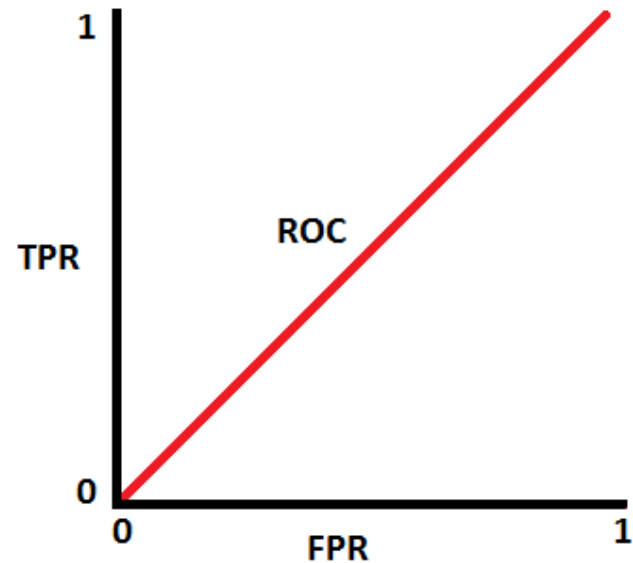
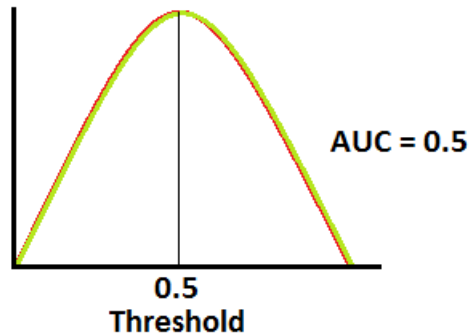
# ROC curves with Type I and II errors

- Red distribution curve is of the positive class
- Green distribution curve is of negative class
- When there is Type 1 and Type II errors



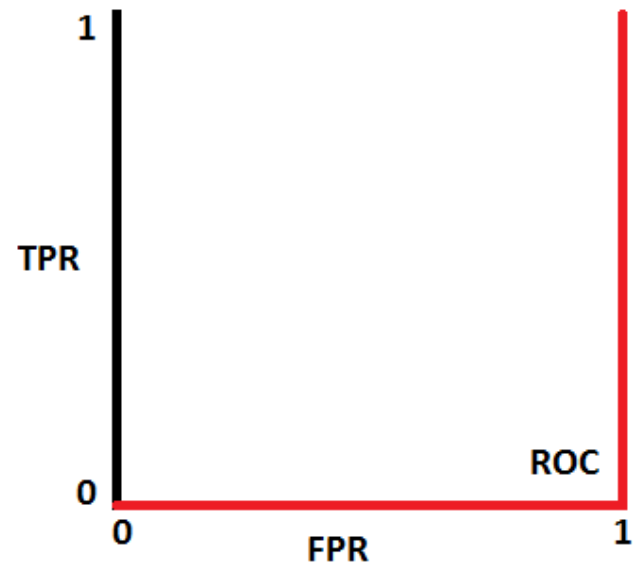
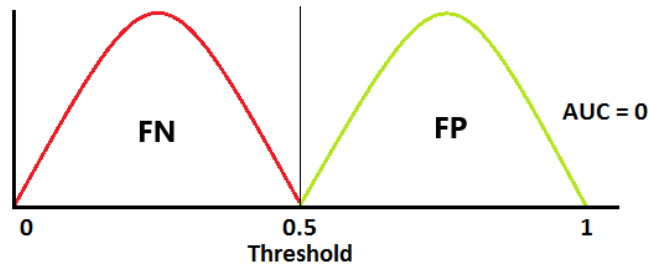
# ROC curves for a random case

- Red distribution curve is of the positive class
- Green distribution curve is of negative class
- The base case: 50% true or false



# ROC curves when perfectly incorrect

- Red distribution curve is of the positive class
- Green distribution curve is of negative class
- The worse case: completely false



# ROC AUC

- The curve gives rise to a useful statistics: ROC AUC
  - AUC = Area Under the Curve
- Ranges from 0 (perfectly incorrect) to 1 (perfectly correct)
- Above 0.6 is generally the minimum acceptable bound
  - 0.7 is preferred and 0.8 is very good
- **package: ROCR** can calculate this too

```
auc_Z <- performance(ROCPred_Z, measure = "auc")  
auc_Z@y.values[[1]]
```

```
## [1] 0.8280943
```

- Note: The objects made by ROCR are not lists!
  - They are *S4 objects*: the 4th version of S (incl. R and S-plus)
  - This is why we use @ to pull out values, not \$
    - That's the only difference you need to know here

# R Practice ROC AUC

- Practice using these new functions with Walmart data
  1. Model decreases in revenue using prior quarter YoY revenue growth
  2. Explore the model using `predict()`
  3. Calculate ROC AUC
  4. Plot an ROC curve
- Do all exercises in today's practice file
  - **R Practice**

# **Academic models: Distance to default (DD)**

# Where does the model come from?

- Merton 1974, Journal of Finance
- Another seminal paper in finance, cited by over 12,000 other academic papers
- Robert C. Merton: 1997 Nobel Prize Winner
  - About Merton

## ON THE PRICING OF CORPORATE DEBT: THE RISK STRUCTURE OF INTEREST RATES\*

ROBERT C. MERTON\*

### I. INTRODUCTION

THE VALUE of a particular issue of corporate debt depends essentially on three items: (1) the required rate of return on riskless (in terms of default) debt (e.g., government bonds or very high grade corporate bonds); (2) the various provisions and restrictions contained in the indenture (e.g., maturity date, coupon rate, call terms, seniority in the event of default, sinking fund, etc.); (3) the probability that the firm will be unable to satisfy some or all of the indenture requirements (i.e., the probability of default).

While a number of theories and empirical studies has been published on the term structure of interest rates (item 1), there has been no systematic development of a theory for pricing bonds when there is a significant probability of default. The purpose of this paper is to present such a theory which might be called a theory of the risk structure of interest rates. The use of the term "risk" is restricted to the possible gains or losses to bondholders as a result of (unanticipated) changes in the probability of default and does not include the gains or losses inherent to all bonds caused by (unanticipated) changes in interest rates in general. Throughout most of the analysis, a given term structure is assumed and hence, the price differentials among bonds will be solely caused by differences in the probability of default.

In a seminal paper, Black and Scholes [1] present a complete general equilibrium theory of option pricing which is particularly attractive because the final formula is a function of "observable" variables. Therefore, the model is subject to direct empirical tests which they [2] performed with some success. Merton [5] clarified and extended the Black-Scholes model. While options are highly specialized and relatively unimportant financial instruments, both Black and Scholes [1] and Merton [5, 6] recognized that the same basic approach could be applied in developing a pricing theory for corporate liabilities in general.

In Section II of the paper, the basic equation for the pricing of financial instruments is developed along Black-Scholes lines. In Section III, the model is applied to the simplest form of corporate debt, the discount bond where no coupon payments are made, and a formula for computing the risk structure of interest rates is presented. In Section IV, comparative statics are used to develop graphs of the risk structure, and the question of whether the term premium is an adequate measure of the risk of a bond is answered. In Section V, the validity in the presence of bankruptcy of the famous Modigliani-Miller

\* Associate Professor of Finance, Massachusetts Institute of Technology. I thank J. Ingersoll for doing the computer simulations and for general scientific assistance. Aid from the National Science Foundation is gratefully acknowledged.

# What is the model about?

- The model itself comes from thinking of debt in an options pricing framework
- Uses the Black-Scholes model to price out a company
- Consider a company to be bankrupt when the company is not worth more than the the debt itself, in expectation

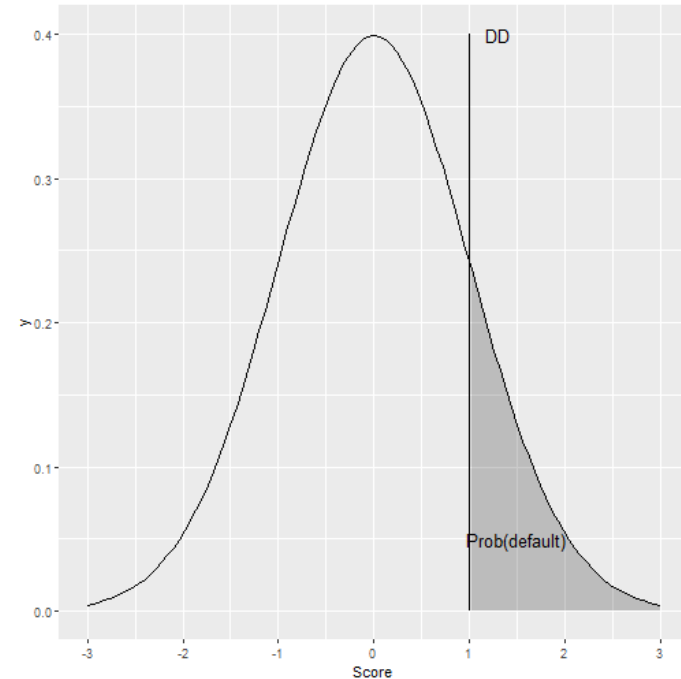
As the name suggests, DD measures the distance to default. It means the higher the DD is, the further away from default. So it is expected to have a negative association between DD and probability of default/bankruptcy.



# Model specification

$$DD = \frac{\log(V_A/D) + (r - (\frac{1}{2}\sigma_A^2))(T - t)}{\sigma_A\sqrt{(T - t)}}$$

- $V_A$ : Value of net assets
  - Market based
- $D$ : Value of liabilities
  - From balance sheet
- $r$ : The annual risk free rate
- $\sigma_A$ : Volatility of assets
  - Use daily stock return volatility, annualized
    - Annualized means multiply by  $\sqrt{252}$
- $T - t$ : Time horizon, taking 252 trading days



# Who uses it?

- Moody's credit risk model is derived from the Merton model
  - Common platform for analyzing risk in financial services
  - **More information**

# MOODY'S ANALYTICS

# Applying DD

# Calculating DD in R

- First we need one more measure: the standard deviation of assets
  - This varies by time, and construction of it is subjective
  - We will use standard deviation over the last 5 years

```
# df_stock is an already prepped csv from CRSP data  
df_stock$date <- as.Date(df_stock$date)  
df <- left_join(df, df_stock[, c("gvkey", "date", "ret", "ret.sd")])
```

```
## Joining, by = c("gvkey", "date")
```

# Calculating DD in R

```
df_rf$date <- as.Date(df_rf$dateeff)
df_rf$year <- year(df_rf$date)
df_rf$month <- month(df_rf$date)

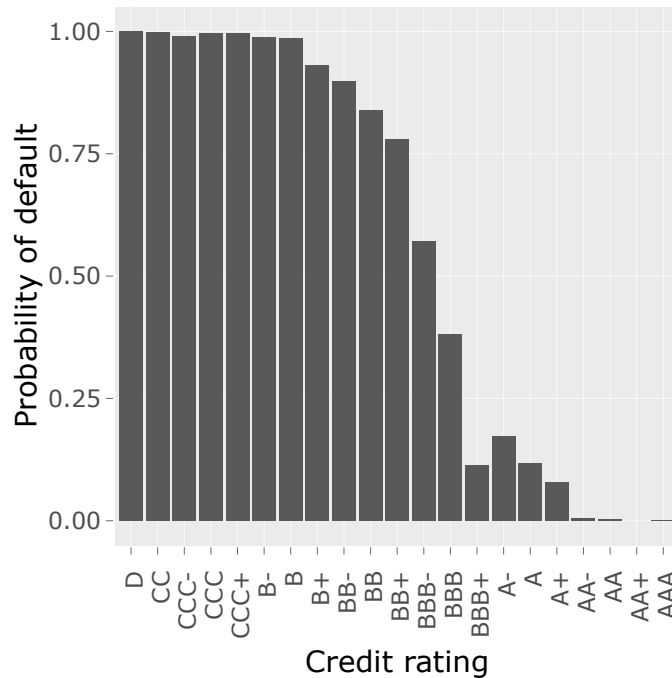
df <- left_join(df, df_rf[, c("year", "month", "rf")])
```

```
## Joining, by = c("year", "month")
```

```
df <- df %>%
  mutate(DD = (log(MVE / lt) + (rf - (ret.sd*sqrt(252))^2 / 2)) /
             (ret.sd * sqrt(252)))
# Clean the measure
df <- df %>%
  mutate_if(is.numeric, list(~replace(., !is.finite(.), NA)))
```

- Just apply the formula using mutate
- $\sqrt{252}$  is included because `ret.sd` is daily return standard deviation
  - There are  $\sim 252$  trading days per year in the US

# DD vs credit ratings, 1973-2017

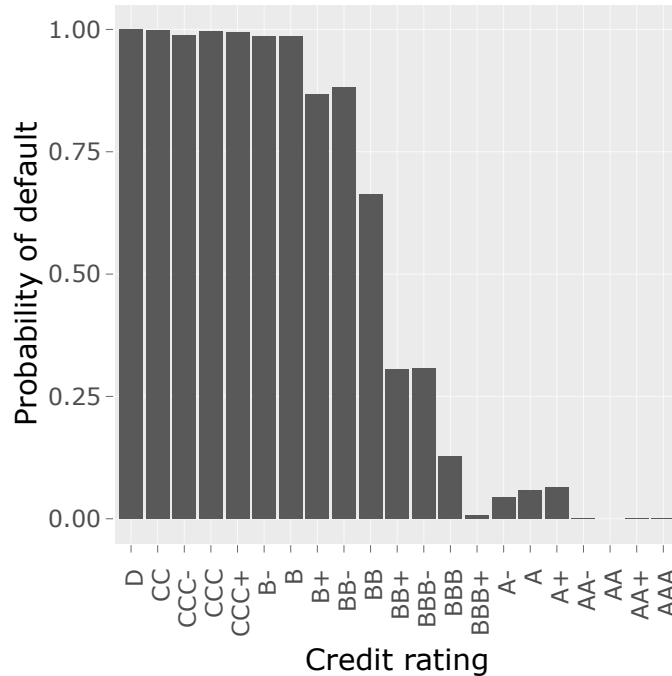


```
df %>%  
  filter(!is.na(DD),  
         !is.na(bankrupt)) %>%  
  group_by(bankrupt) %>%  
  mutate(mean_DD=mean(DD, na.rm=T),  
         prob_default =  
           pnorm(-1 * mean_DD)) %>%  
  slice(1) %>%  
  ungroup() %>%  
  select(bankrupt, mean_DD,  
         prob_default) %>%  
  html_df()
```

bankrupt	mean_DD	prob_default
0	0.612414	0.2701319
1	-2.447382	0.9928051

- **pnorm()** calculates **c.d.f.** of normal distribution (ie, the probability of < DD)

# DD vs credit ratings, 2000-2017



```
df %>%
  filter(!is.na(DD),
         !is.na(bankrupt),
         year >= 2000) %>%
  group_by(bankrupt) %>%
  mutate(mean_DD=mean(DD, na.rm=T),
         prob_default =
           pnorm(-1 * mean_DD)) %>%
  slice(1) %>%
  ungroup() %>%
  select(bankrupt, mean_DD,
         prob_default) %>%
  html_df()
```

bankrupt	mean_DD	prob_default
0	0.8411654	0.2001276
1	-4.3076039	0.9999917

# Test it with a regression

```
fit_DD <- glm(bankrupt ~ DD, data = df, family = binomial)
summary(fit_DD)
```

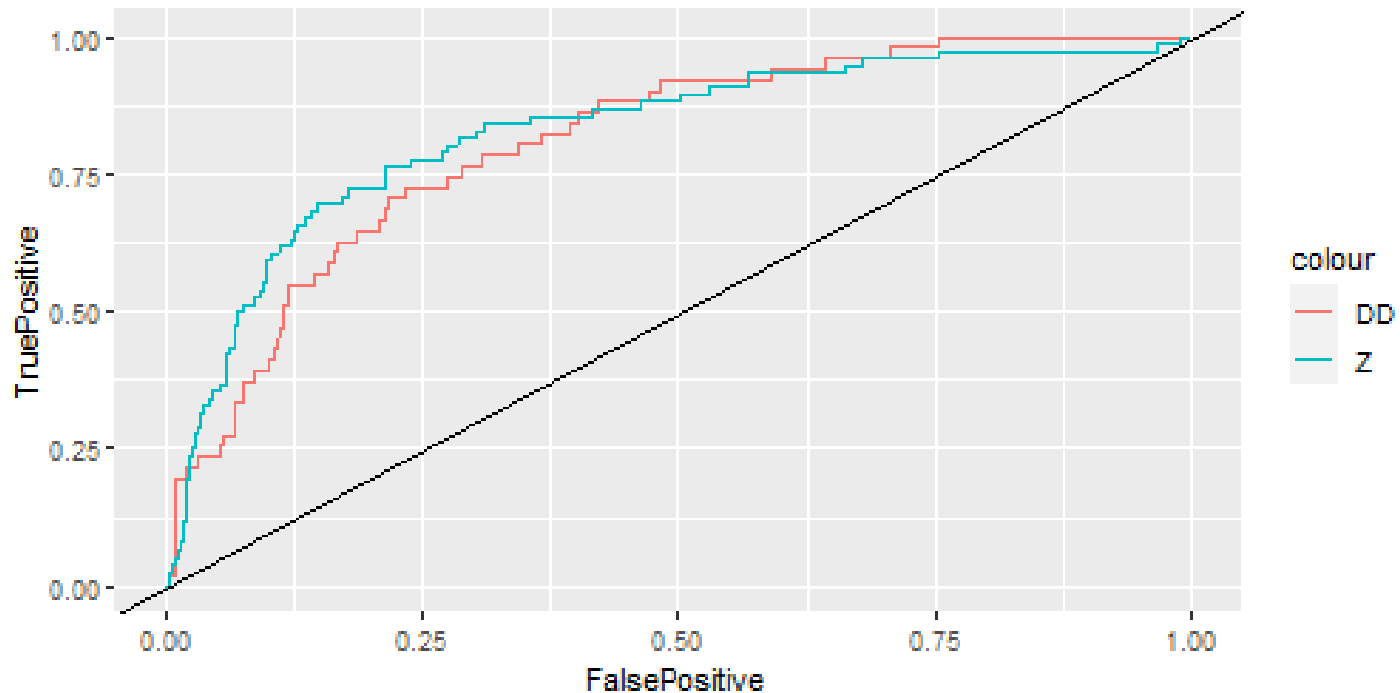
```
##
## Call:
## glm(formula = bankrupt ~ DD, family = binomial, data = df)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9848  -0.0750  -0.0634  -0.0506   3.6506
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.16401    0.15323  -40.23  < 2e-16 ***
## DD          -0.24451    0.03773   -6.48  9.14e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 718.67  on 21563  degrees of freedom
## Residual deviance: 677.18  on 21562  degrees of freedom
## (33618 observations deleted due to missingness)
## AIC: 681.18
##
## Number of Fisher Scoring iterations: 9
```



# ROC Curves

```
dfDD <- df %>% filter(!is.na(DD), !is.na(bankrupt))
pred_DD <- predict(fit_DD, dfDD, type = "response")
ROCpred_DD <- prediction(as.numeric(pred_DD), as.numeric(dfDD$bankrupt))
ROCperf_DD <- performance(ROCpred_DD, 'tpr', 'fpr')
df_ROC_DD <- data.frame(FalsePositive=c(ROCperf_DD@x.values[[1]]),
                        TruePositive=c(ROCperf_DD@y.values[[1]]))

ggplot() +
  geom_line(data=df_ROC_DD, aes(x=FalsePositive, y=TruePositive, color="DD")) +
  geom_line(data=df_ROC_Z, aes(x=FP, y=TP, color="Z")) + geom_abline(slope=1)
```



# AUC comparison

```
#AUC
auc_DD <- performance(ROCpred_DD, measure = "auc")
AUCs <- c(auc_Z@y.values[[1]], auc_DD@y.values[[1]])
names(AUCs) <- c("Z", "DD")
AUCs
```

```
##           Z           DD
## 0.8280943 0.8097803
```

Both measures perform similarly, but Altman Z performs slightly better.

**A more practical application**

# A more practical application

- Companies don't only have problems when there is a bankruptcy
  - Credit downgrades can be just as bad

| Why?

- Credit downgrades cause an increase in interest rates for debt, leading to potential liquidity issues.

# Predicting downgrades

```
# calculate downgrade
df <- df %>% arrange(gvkey, date) %>%
  group_by(gvkey) %>%
  mutate(downgrade = ifelse(rating < lag(rating), 1, 0))

# training sample
train <- df %>% filter(year < 2015)
test <- df %>% filter(year >= 2015)

# glms
fit_Z2 <- glm(downgrade ~ Z, data = train, family = binomial)
fit_DD2 <- glm(downgrade ~ DD, data = train, family = binomial)
```

# Predicting downgrades with Z

```
summary(fit_Z2)
```

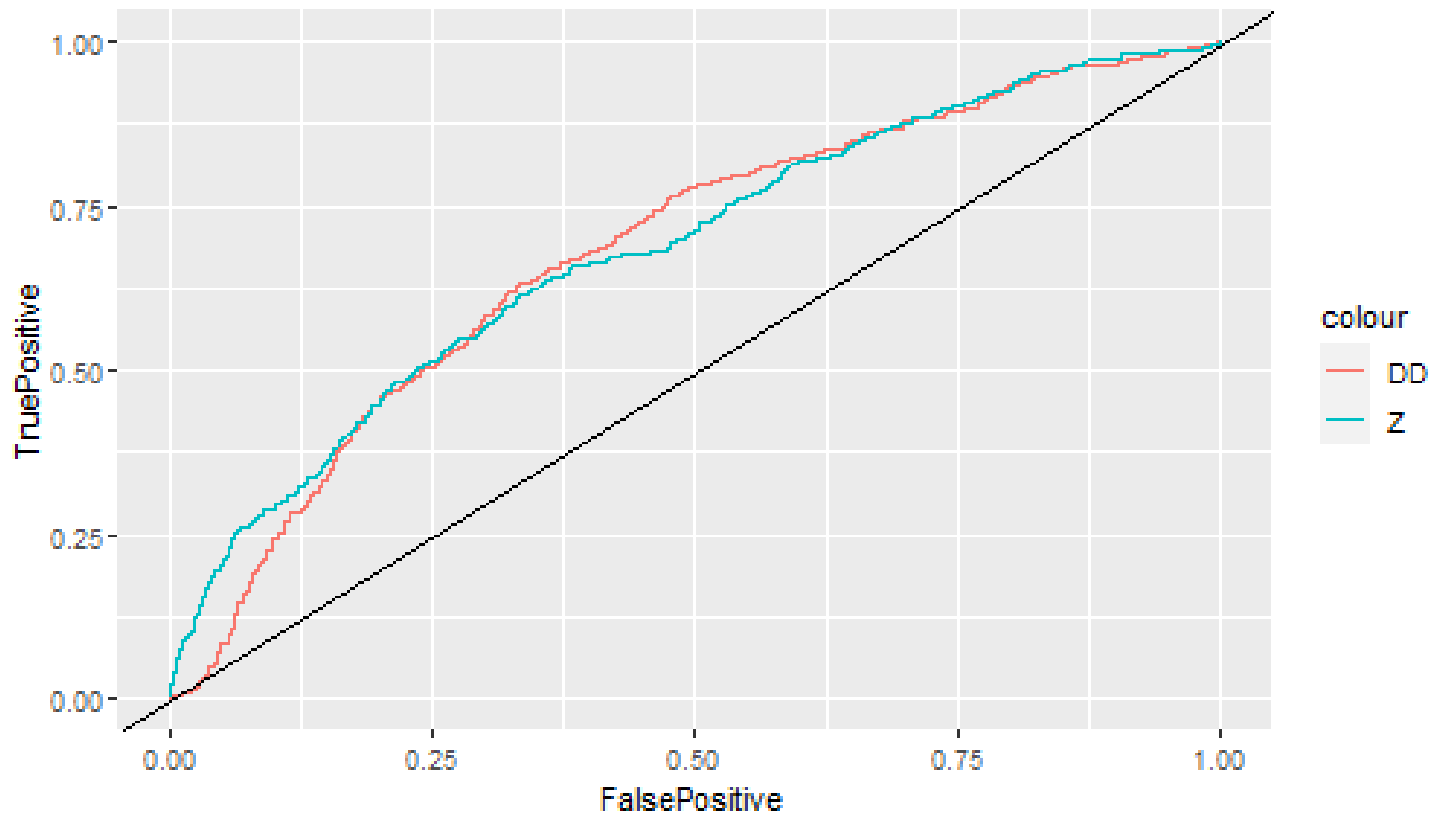
```
##  
## Call:  
## glm(formula = downgrade ~ Z, family = binomial, data = train)  
##  
## Deviance Residuals:  
##      Min       1Q   Median       3Q      Max   
## -1.1223  -0.5156  -0.4418  -0.3277   6.4638   
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)      
## (Intercept) -1.10377    0.09288  -11.88  <2e-16 ***   
## Z            -0.43729    0.03839  -11.39  <2e-16 ***   
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
##      Null deviance: 3874.5  on 5795  degrees of freedom  
## Residual deviance: 3720.4  on 5794  degrees of freedom  
## (47058 observations deleted due to missingness)  
## AIC: 3724.4  
##  
## Number of Fisher Scoring iterations: 6
```

# Predicting downgrades with DD

```
summary(fit_DD2)
```

```
##  
## Call:  
## glm(formula = downgrade ~ DD, family = binomial, data = train)  
##  
## Deviance Residuals:  
##      Min       1Q   Median       3Q      Max   
## -1.7319  -0.5004  -0.4278  -0.3343   3.0755   
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)      
## (Intercept) -2.36365    0.05607  -42.15  <2e-16 ***   
## DD          -0.22224    0.02035  -10.92  <2e-16 ***   
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
##      Null deviance: 3115.3  on 4732  degrees of freedom  
## Residual deviance: 2982.9  on 4731  degrees of freedom  
## (48121 observations deleted due to missingness)  
## AIC: 2986.9  
##  
## Number of Fisher Scoring iterations: 5
```

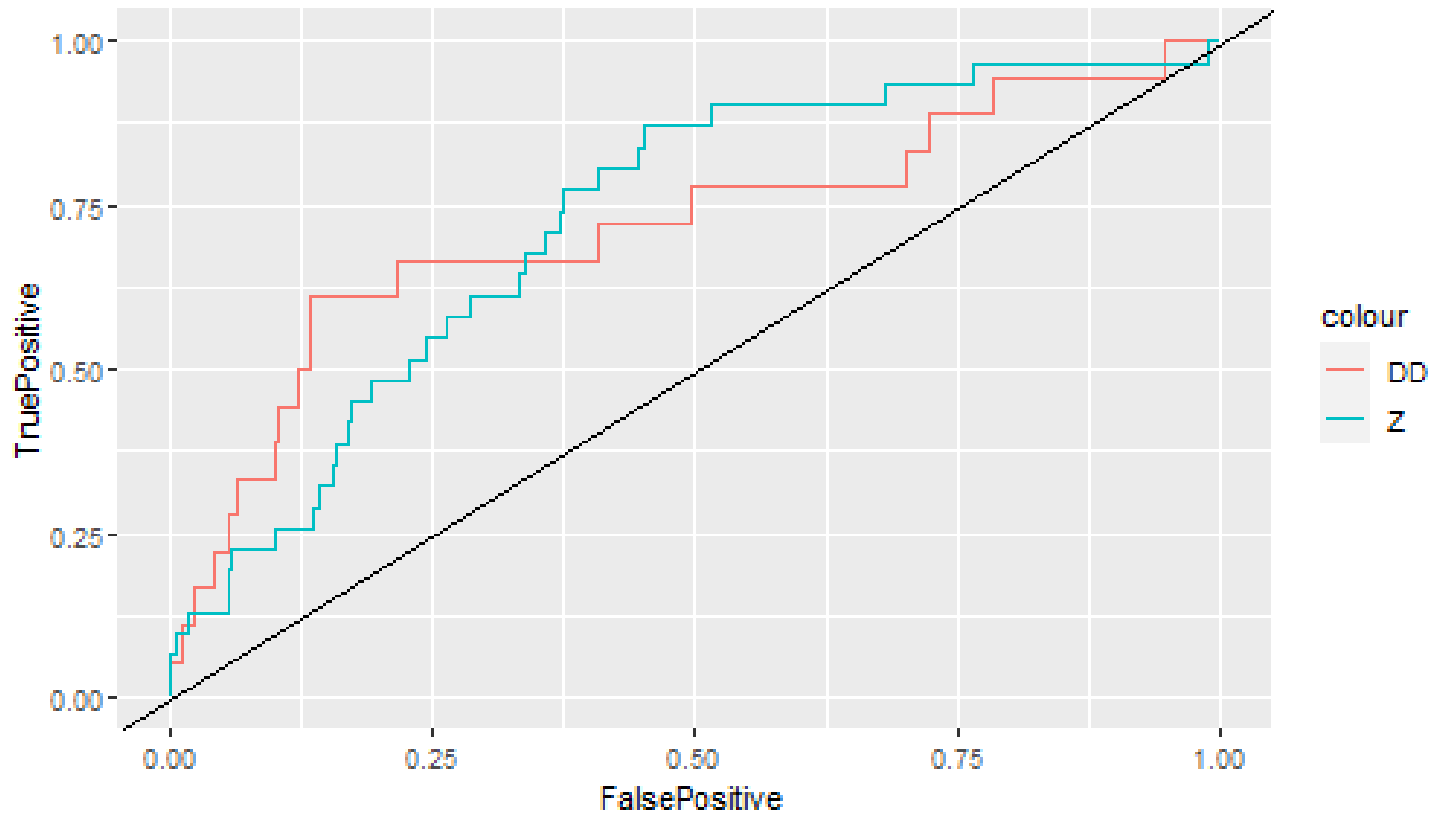
# ROC Performance on this task



```
##          Z          DD
## 0.6839086 0.6811973
```



# Out of sample ROC performance



```
##          Z          DD
## 0.7270046 0.7183575
```

# Predicting bankruptcy

What other data could we use to predict corporate bankruptcy as it relates to a company's supply chain?

- What is the reason that this event or data would be useful for prediction?
  - i.e., how does it fit into your mental model?
- A useful starting point from McKinsey
  - **Big data and the supply chain**
    - Section "B. Sourcing"

# Summary of Session 8

# For next week


- Try to replicate the code
- Continue your Datacamp career track
- Have you submitted to Kaggle/Tianchi to check your model performance?

# R Coding Style Guide

Style is subjective and arbitrary but it is important to follow a generally accepted style if you want to share code with others. I suggest the [The tidyverse style guide](#) which is also adopted by [Google](#) with some modification

- Highlights of **the tidyverse style guide**:
  - *File names*: end with .R
  - *Identifiers*: variable\_name, function\_name, try not to use "." as it is reserved by Base R's S3 objects
  - *Line length*: 80 characters
  - *Indentation*: two spaces, no tabs (RStudio by default converts tabs to spaces and you may change under global options)
  - *Spacing*: `x = 0`, not `x=0`, no space before a comma, but always place one after a comma
  - *Curly braces {}*: first on same line, last on own line
  - *Assignment*: use `<-`, not `=` nor `->`
  - *Semicolon(,)*: don't use, I used once for the interest of space
  - *return()*: Use explicit returns in functions: default function return is the last evaluated expression
  - *File paths*: use **relative file path** `"../..filename.csv"` rather than absolute path `"C:/mydata/filename.csv"`. Backslash needs `\\`

# R packages used in this slide

This slide was prepared on 2021-09-08 from Session\_8s.Rmd with R version 4.1.1 (2021-08-10) Kick Things on Windows 10 x64 build 18362 .

The attached packages used in this slide are:

```
##      ROCR      EnvStats  lubridate    plotly      forcats    stringr      dplyr
## "1.0-11"    "2.4.0"    "1.7.10"   "4.9.4.1"   "0.5.1"    "1.4.0"      "1.0.7"
##      purrr      readr      tidyr      tibble      ggplot2    tidyverse    kableExtra
## "0.3.4"    "2.0.1"    "1.1.3"    "3.1.3"     "3.3.5"    "1.3.1"      "1.3.4"
##      knitr
## "1.33"
```