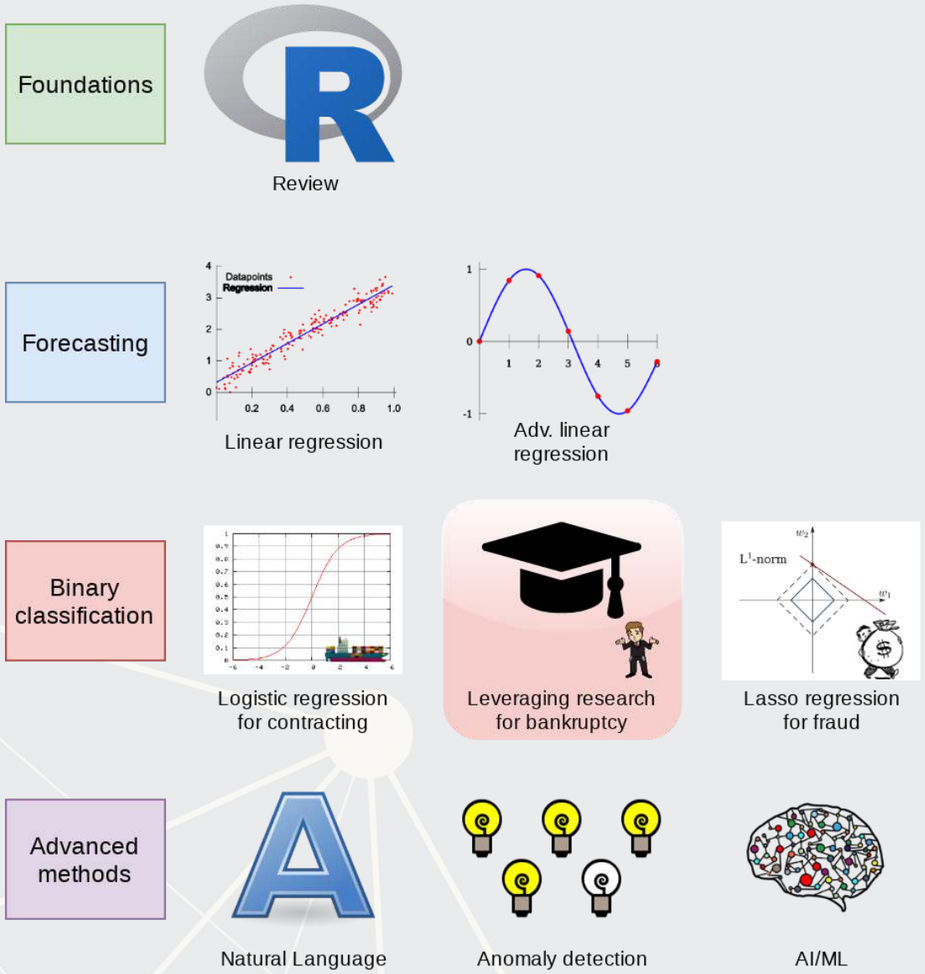
ACCT 420: Logistic Regression for Bankruptcy

Session 5

Dr. Richard M. Crowley

Front matter

Learning objectives



- **Theory:**
  - Academic research
- **Application:**
  - Predicting bankruptcy over the next year for US manufacturing firms
    - Extend to credit downgrades
- **Methodology:**
  - Logistic regression
  - Models from academic research

Datacamp

- Explore on your own
- No specific required class this week

Academic research

History of academic research in accounting

- Academic research in accounting, as it is today, began in the 1960s
  - What we call *Positive Accounting Theory*
    - Positive theory: understanding how the world works
- Prior to the 1960s, the focus was on Prescriptive theory
  - How the world should work
- Accounting research builds on work from many fields:
  - Economics
  - Finance
  - Psychology
  - Econometrics
  - Computer science (more recently)

Types of academic research

- Theory
  - Pure economics proofs and simulation
- Experimental
  - Proper experimentation done on individuals
  - Can be psychology experiments or economic experiments
- Empirical/Archival
  - Data driven research
  - Based on the usage of historical data (i.e., archives)
  - Most likely to be easily co-optable by businesses and regulators

Who leverages accounting research

- Hedge funds
- Mutual funds
- Auditors
- Law firms
- Government entities like SG MAS and US SEC
- Exchanges like SGX

Where can you find academic research

- The SMU library has access to almost all high quality accounting research
- Google Scholar is a great site to discover research past and present
- SSRN is the site to find cutting edge accounting and business research
  - List of top accounting papers on SSRN (by downloads)

Academic models: Altman Z-Score

First: Why care about bankruptcy?

- Read this article: rmc.link/420class5-1
  - “Carillion’s liquidation reveals the dangers of shared sourcing”

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

Where does the model come from?

- Altman 1968, Journal of Finance
- A seminal paper in Finance cited over 15,000 times by other academic papers

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.[1] 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.

What is the model about?

- The model was developed to identify firms that are likely to go bankrupt out of a pool of firms
- Focuses on using ratio analysis to determine such firms

Model specification

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

- x_1: Working capital to assets ratio
- x_2: Retained earnings to assets ratio
- x_3: EBIT to assets ratio
- x_4: Market value of equity to book value of liabilities
- x_5: Sales to 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
- Used data from 1946 through 1965
  - 33 US manufacturing firms that went bankrupt, 33 that survived

More about this, from Altman himself in 2000: rmc.link/420class5-2

- Read the section "Variable Selection" starting on page 8
  - Skim through x_1, x_2, x_3, x_4, and x_5

> How would these assumptions stand today? Rate at wooclap.com/420W5

Who uses Altman Z?

- 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

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?

Question structure

> 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`
- Most 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 in the US

- Chapter 7
  - The company ceases operating and liquidates
- Chapter 11
  - For firms intending to reorganize the company to “try to become profitable again” (US SEC)

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
# 100338 is an outlier in the bonds distribution
df <- df %>% filter(at >= 1, revt >= 1, gvkey != 100338)

## Merge in stock value
df$date <- as.Date(df$datadate)
df_mve <- df_mve %>%
  mutate(date = as.Date(datadate),
         mve = csho * prcc_f) %>%
  rename(gvkey=GVKEY)

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

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

```
df <- df %>%
  group_by(gvkey) %>%
  mutate(bankrupt = ifelse(row_number() == n() & dlrsn == 2 &
                           !is.na(dlrsn), 1, 0)) %>%
  ungroup()
```

- `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

```r
# Calculate the measures needed
df <- df %>%
  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 %>%
  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

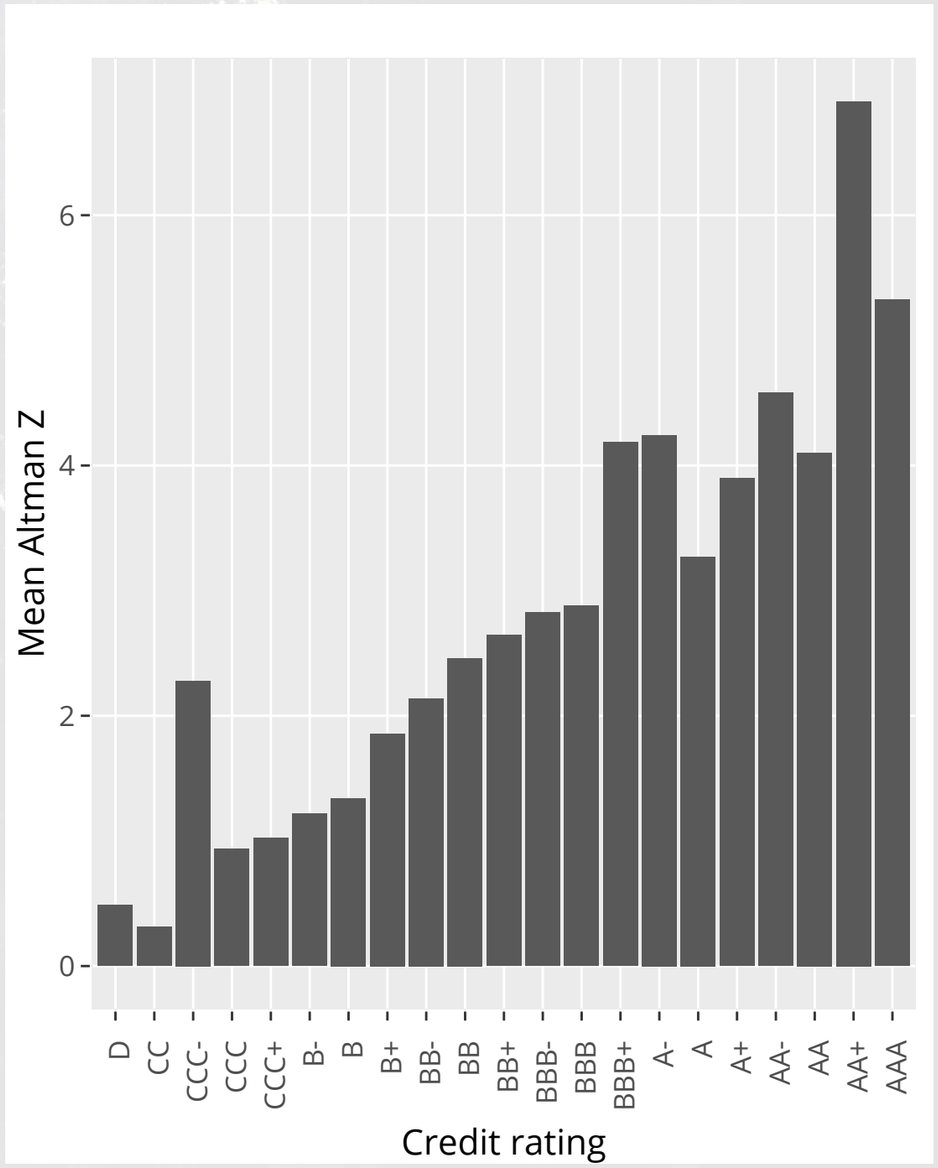
```
# df_ratings has ratings data in it

# 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 numeric 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()
```

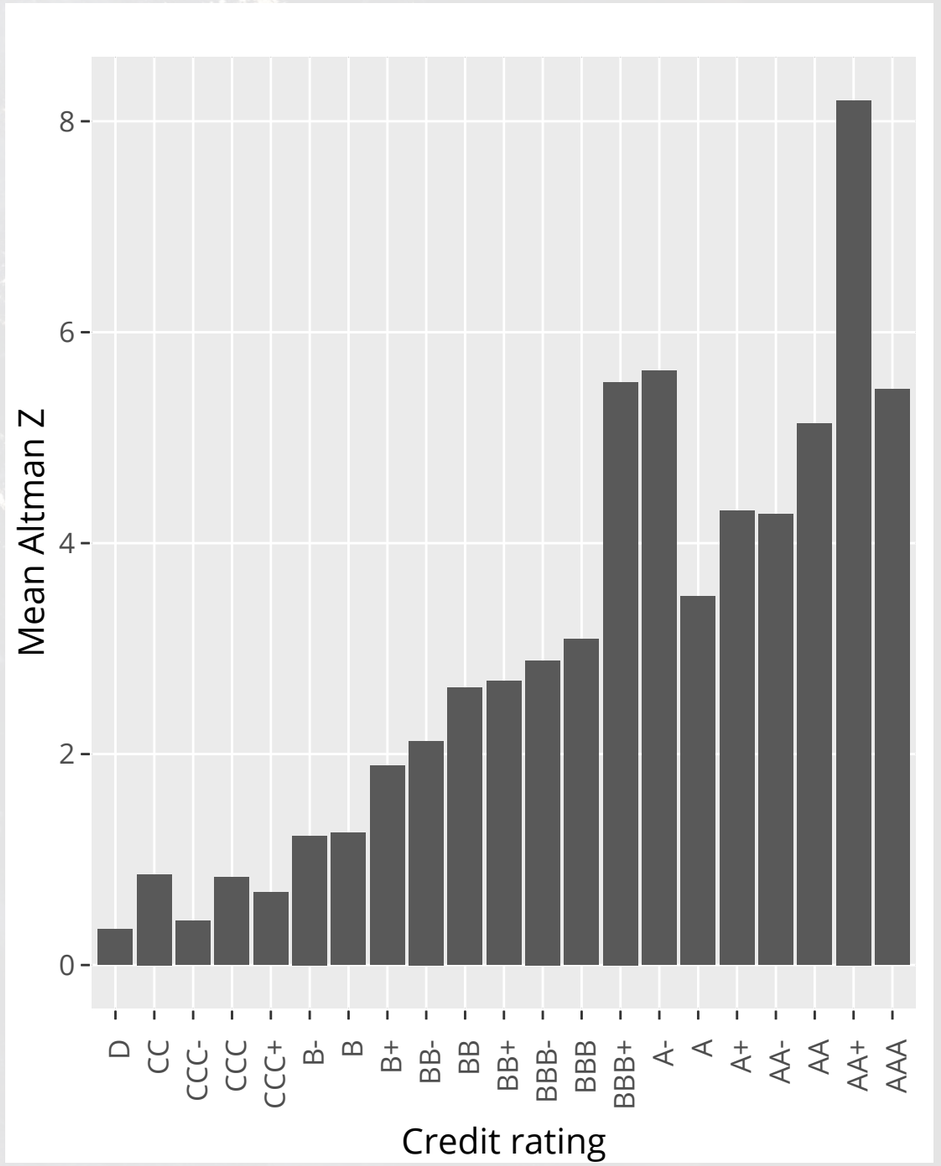
bankrupt	mean_Z
0	3.939223
1	0.927843



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
```

How good is the model though???

Examples:

> Correct 92.0% of the time using Z < 1 as a cutoff

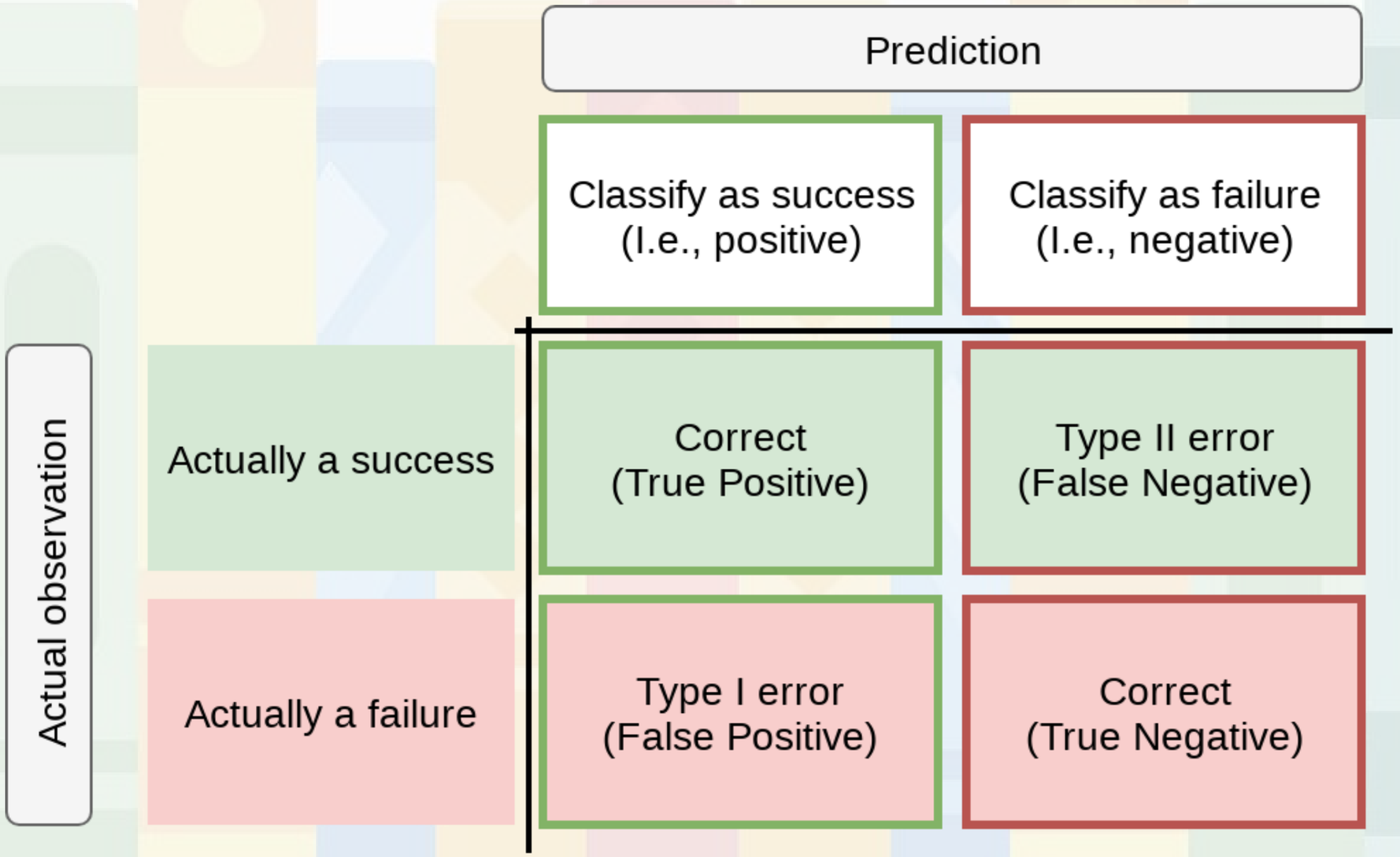
- Correctly captures 39 of 83 bankruptcies

> Correct 99.7% of the time if we say firms never go bankrupt…

- Correctly captures **0** of 83 bankruptcies

Errors in binary testing

Types of errors



> This type of chart (filled in) is called a *Confusion matrix*

Type I error (False positive)

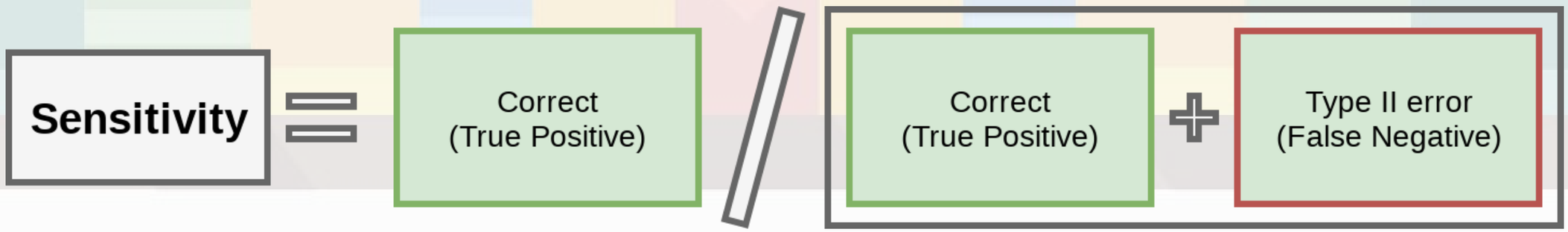
> We say that the company will go bankrupt, but they don't

- 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): $1 - Sensitivity$
    - The percent of successes misclassified as failures
  - Sensitivity:
    - A.k.a. true positive rate (TPR)
    - The percent of successes properly classified



Useful equations

FPR = Type I error (False Positive) / (Type I error (False Positive) + Correct (True Negative))

Sensitivity = Correct (True Positive) / (Correct (True Positive) + Type II error (False Negative))

Specificity = Correct (True Negative) / (Type I error (False Positive) + Correct (True Negative))

Accuracy = (Correct (True Positive) + Correct (True Negative)) / (Correct (True Positive) + Type II error (False Negative) + Type I error (False Positive) + Correct (True Negative))

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
- Sensitivity is very useful for rare events
- Specificity is very useful for frequent events
  - Or for events where misclassifying the null is very troublesome
    - Criminal trials
    - Medical diagnoses

Let's plot TPR and FPR out

- `ROCR` can calculate these for us!

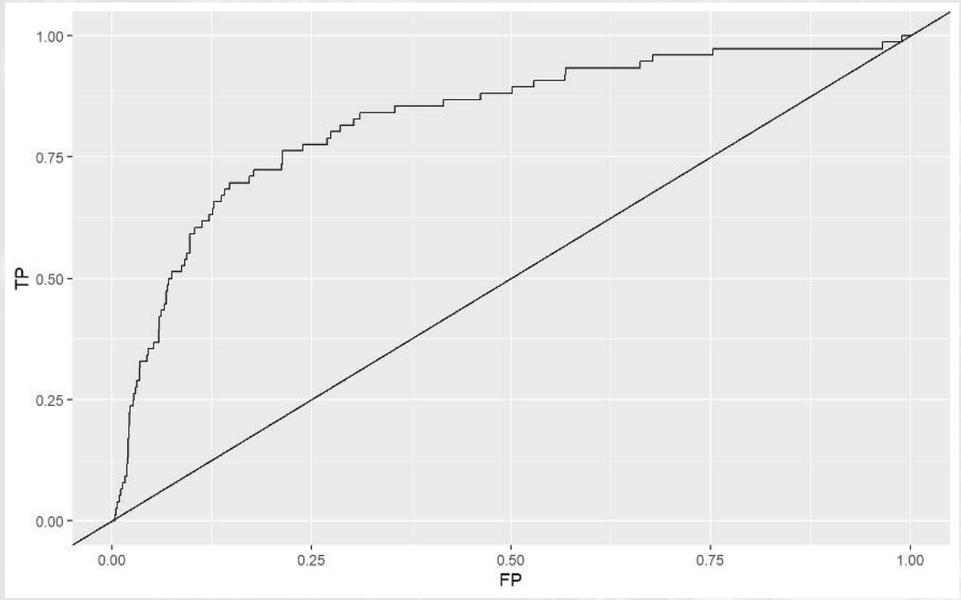
```
library(ROCR)
dfZ <- df %>% filter(!is.na(Z), !is.na(bankrupt))
pred_Z <- predict(fit_Z, dfZ, type="response")
ROCpred_Z <- prediction(as.numeric(pred_Z), as.numeric(dfZ$bankrupt))
ROCperf_Z <- performance(ROCpred_Z, 'tpr','fpr')
```

- Notes on `ROCR`:
  1. The functions are rather picky and fragile. Likely sources of error include:
     - 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()` calculates performance measures
     - It knows 35 of them
     - `'tpr'` is true positive rate
     - `'fpr'` is false positive rate

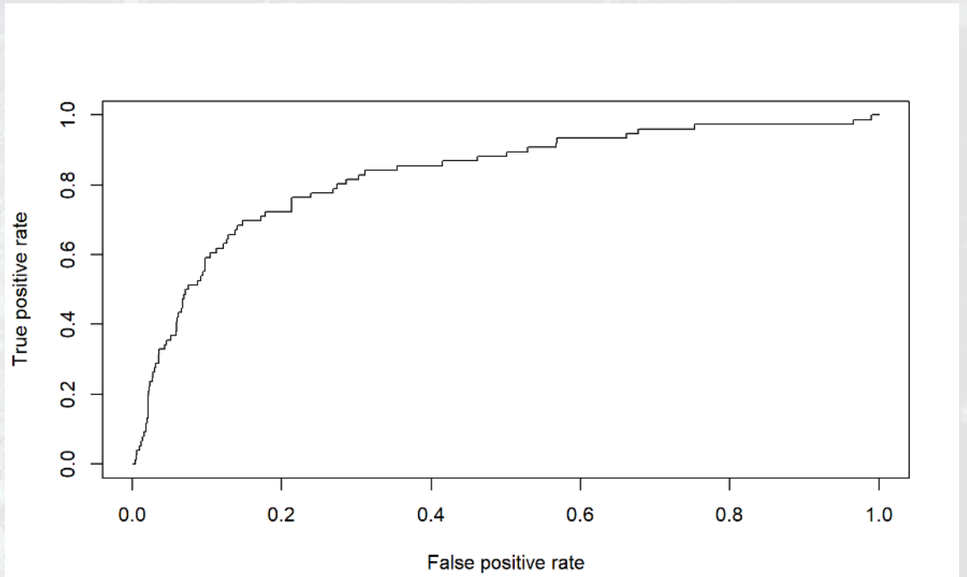
Let's plot TPR and FPR out

- 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)
```

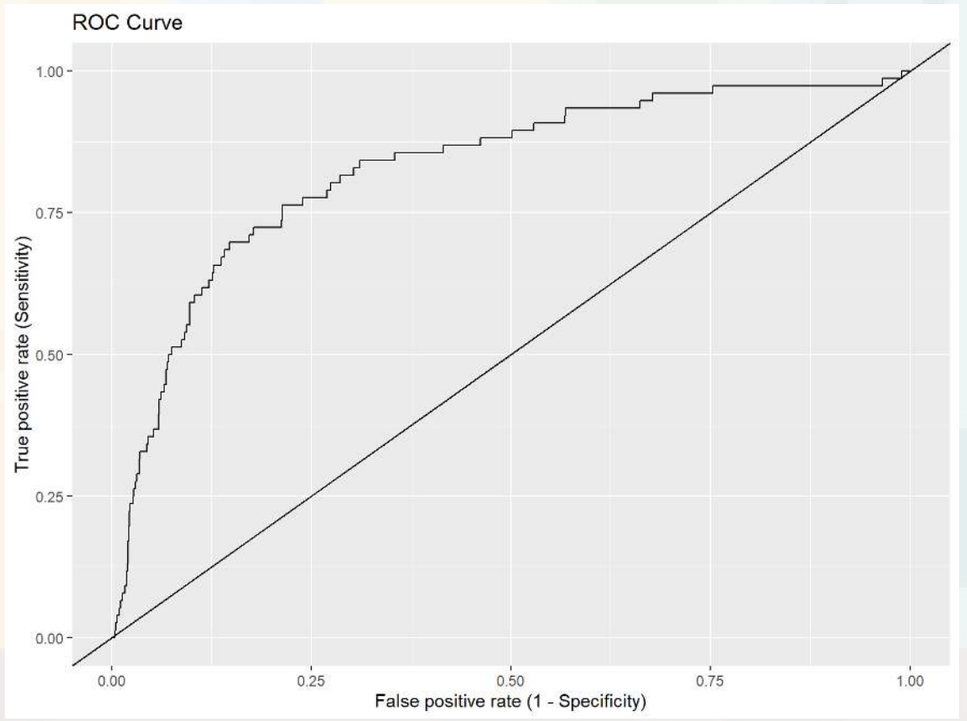



```
plot(ROCperf_Z)
```



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 logistic regression fits.
  - Neat properties:
    - The area under a perfect model is always 1
    - The area under random chance is always 0.5
      - This is the straight line on the graph



ROC AUC

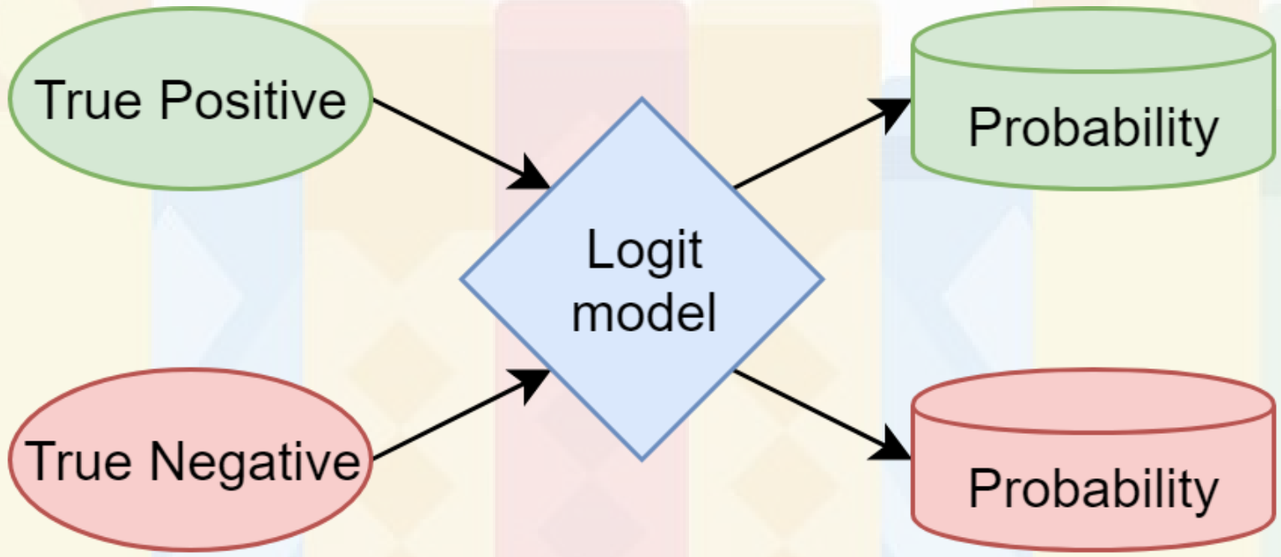
- The neat properties of the curve give rise to a useful statistic: ROC AUC
  - AUC = Area under the curve
- Ranges from 0 (perfectly incorrect) to 1 (perfectly correct)
- Above 0.6 is generally the minimum acceptable bound
  - 0.7 is preferred
  - 0.8 is very good
- `ROCR` can calculate this too

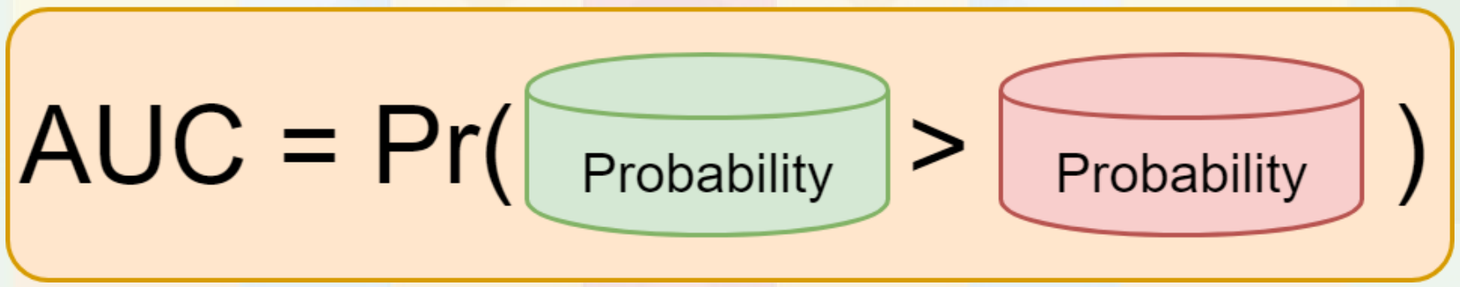
```
auc_Z <- performance(ROCpred_Z, measure = "auc")
auc_Z@y.values[[1]]
```

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

  - Note: The objects made by ROCR are not lists!
    - They are “S4 objects”
    - This is why we use `@` to pull out values, not `$`
      - That’s the only difference you need to know here

ROC AUC simplest interpretation





AUC is the probability that our model assigns a higher estimated probability to a randomly selected 1 than to a randomly selected 0.

R Practice ROC AUC

- Practice using these new functions with last week's 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 a ROC curve
- Do all exercises in today's practice file
  - R Practice
  - Shortlink: rmc.link/420r5

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
- 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.

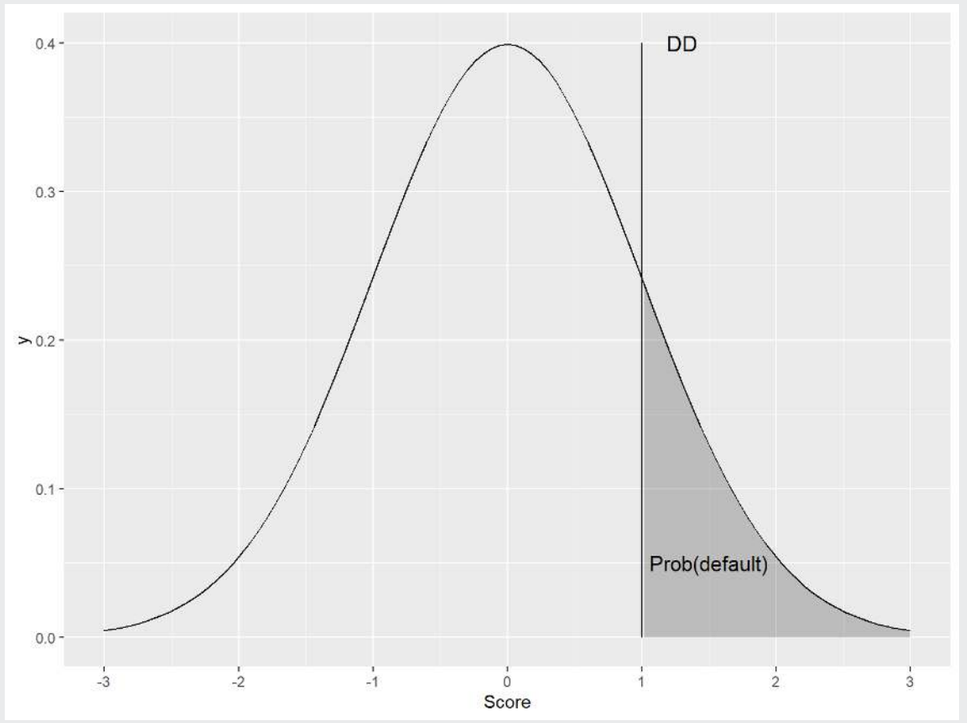
What is the model about?

- 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

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 assets
  - Market based
- D: Value of liabilities
  - From balance sheet
- r: The risk free rate
- σ_A: Volatility of assets
  - Use daily stock return volatility, annualized
    - Annualized means multiply by $\sqrt{253}$
- $T-t$: Time horizon



Who uses it?

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

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$date <- as.Date(df_rf$dateff)
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")
```

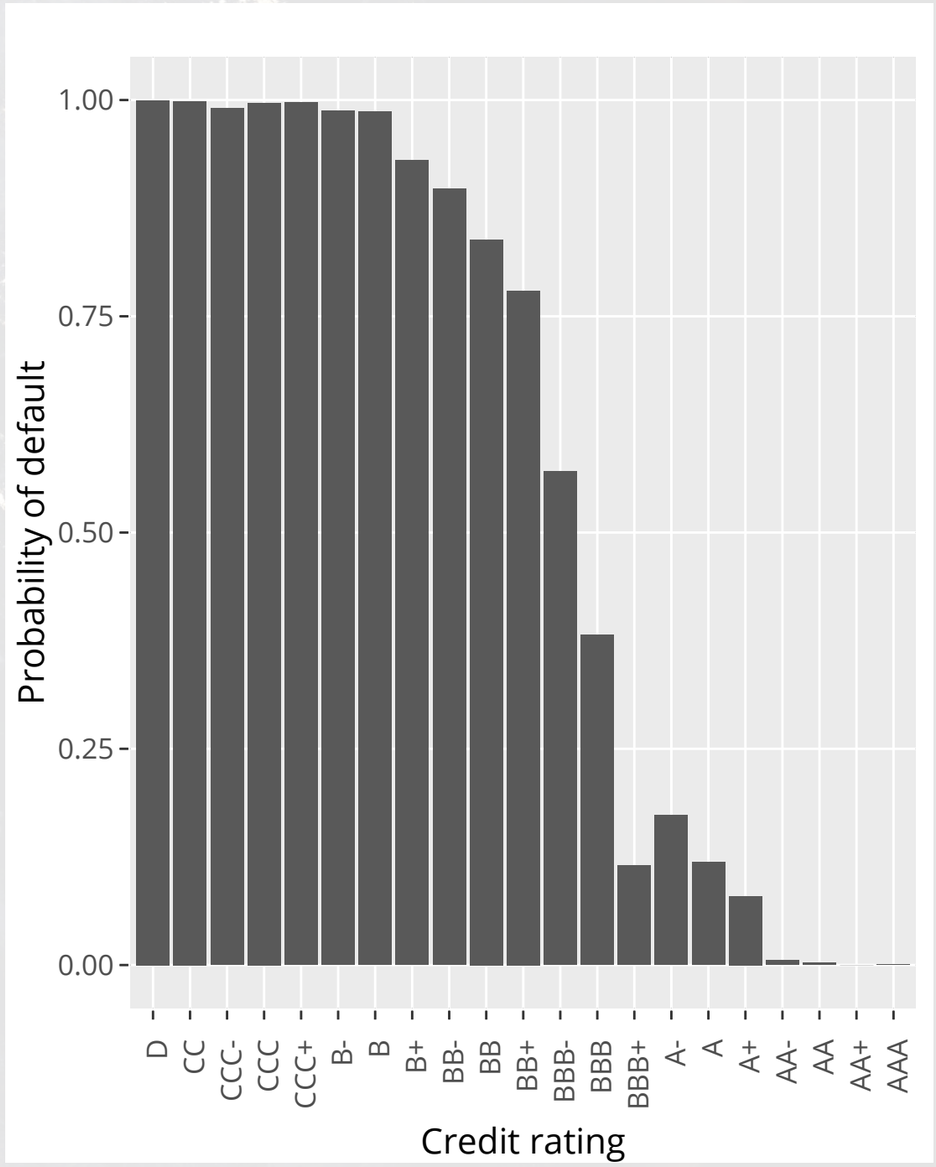
```
df <- df %>%
  mutate(DD = (log(mve / lt) + (rf - (ret.sd*sqrt(253))^2 / 2)) /
                (ret.sd*sqrt(253)))
# Clean the measure
df <- df %>%
  mutate_if(is.numeric, list(~replace(., !is.finite(.), NA)))
```

- Just apply the formula using mutate
- $\sqrt{253}$ is included because `ret.sd` is daily return standard deviation
  - There are ~253 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()
```

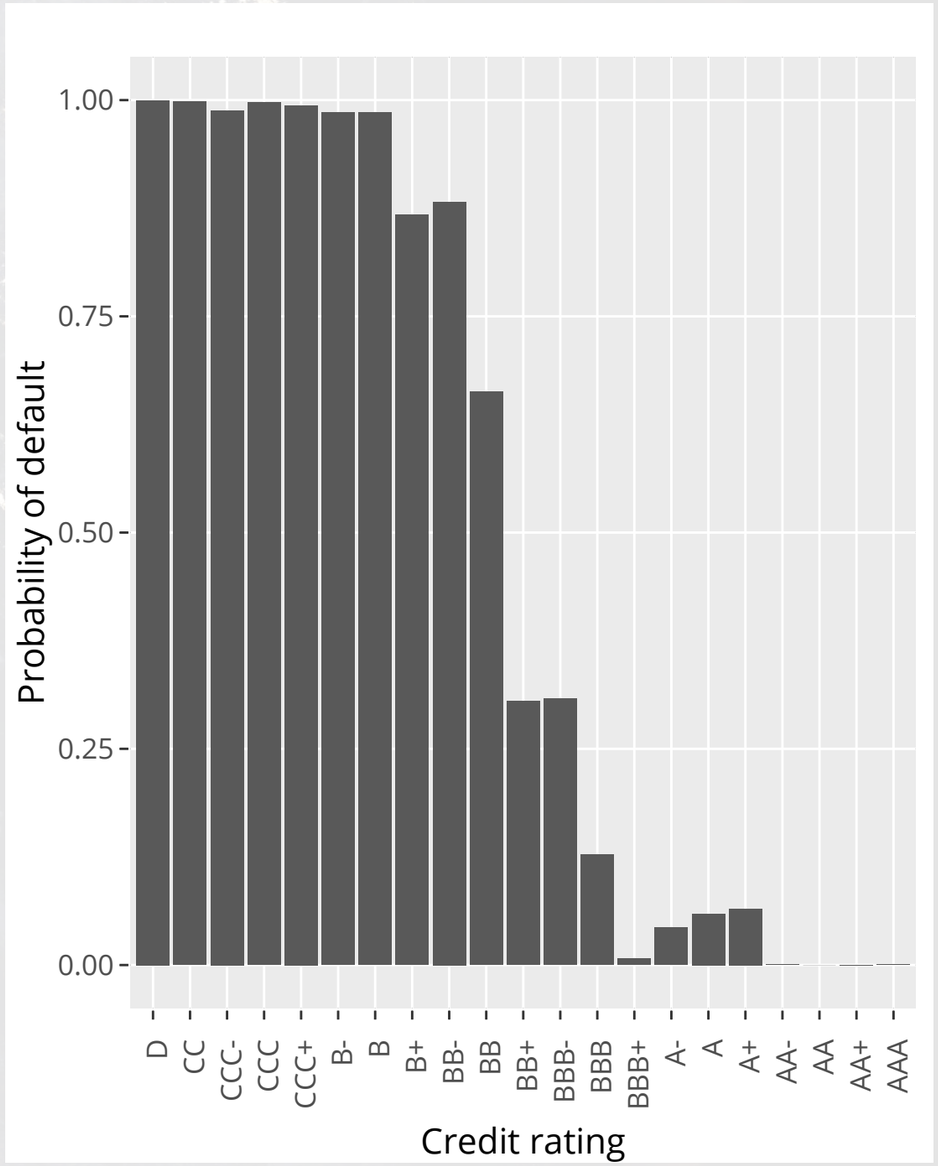
bankrupt	mean_DD	prob_default
0	0.6096854	0.2710351
1	-2.4445081	0.9927475



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	0.8379932	0.2010172
1	-4.3001844	0.9999915



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
## -2.9929  -0.0750  -0.0634  -0.0506   3.6503
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.16394    0.15322 -40.230  < 2e-16 ***
## DD          -0.24459    0.03781  -6.469 9.89e-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: 677.27  on 21562  degrees of freedom
##   (33618 observations deleted due to missingness)
## AIC: 681.27
```

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 0.8098304
```

> 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 %>%
  group_by(gvkey) %>%
  arrange(date) %>%
  mutate(downgrade = ifelse(rating < lag(rating),1,0),
         diff_Z = Z - lag(Z),
         diff_DD = DD - lag(DD)) %>%
  ungroup()


# training sample
train <- df %>% filter(year < 2014, !is.na(diff_Z), !is.na(diff_DD), !is.na(downgr
                       year > 1985)
test <- df %>% filter(year >= 2014, !is.na(diff_Z), !is.na(diff_DD), !is.na(downgr

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

Predicting downgrades with Altman Z

```
summary(fit_Z2)
```

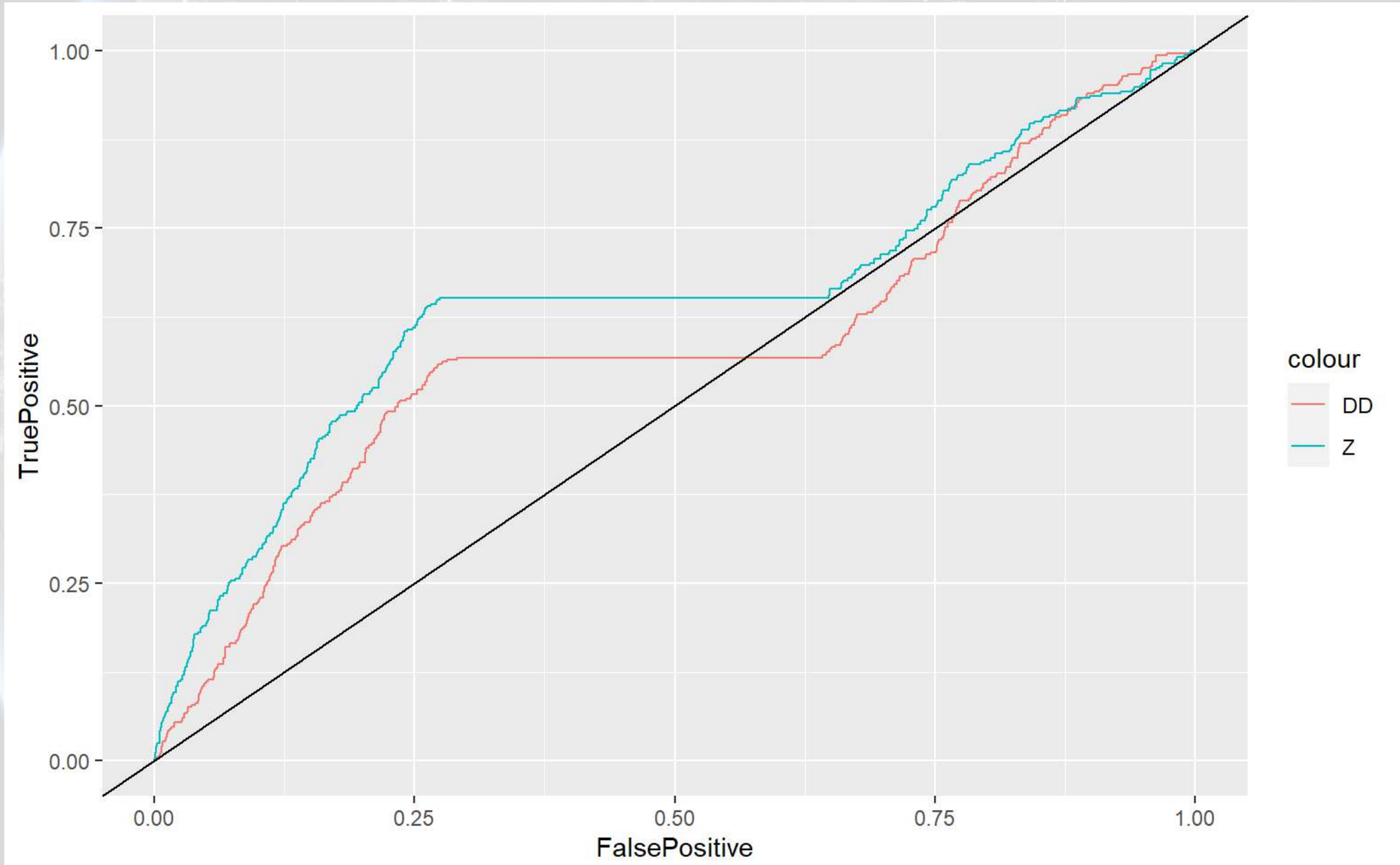
```
##
## Call:
## glm(formula = downgrade ~ diff_Z, family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.4115  -0.4428  -0.4428  -0.3928   2.7437
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.27310    0.06139 -37.029   <2e-16 ***
## diff_Z      -0.77150    0.09245  -8.345   <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: 2145.3  on 3277  degrees of freedom
## Residual deviance: 2065.8  on 3276  degrees of freedom
## AIC: 2069.8
##
```

Predicting downgrades with DD

```
summary(fit_DD2)
```

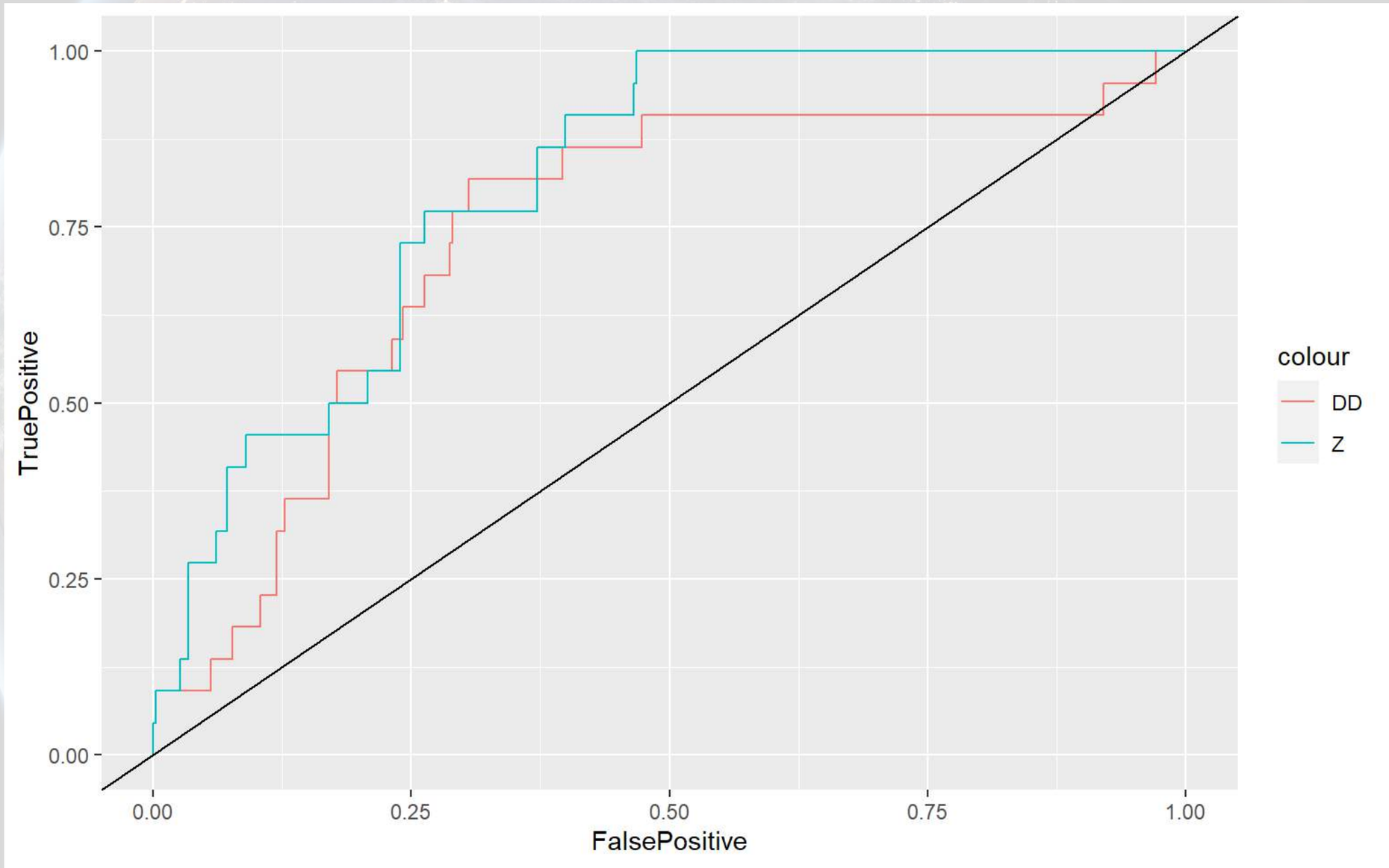
```
##
## Call:
## glm(formula = downgrade ~ diff_DD, family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5726  -0.4565  -0.4558  -0.4095   2.6804
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.21199    0.05926 -37.325  < 2e-16 ***
## diff_DD     -0.21378    0.03723  -5.742 9.37e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2145.3  on 3277  degrees of freedom
## Residual deviance: 2113.2  on 3276  degrees of freedom
## AIC: 2117.2
##
```

ROC Performance on this task




```
##         Z        DD
## 0.6465042 0.5847885
```

Out of sample ROC performance




```
##         Z        DD
## 0.8134671 0.7420213
```

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
  - rmc.link/420class5-3
    - Section "B. Sourcing"

Combined model

Building a combined model

```
fit_comb <- glm(downgrade ~ diff_Z + diff_DD, data=train, family=binomial)
summary(fit_comb)
```

```
##
## Call:
## glm(formula = downgrade ~ diff_Z + diff_DD, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3263  -0.4431  -0.4430  -0.3892   2.7504
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.27217    0.06144 -36.980  < 2e-16 ***
## diff_Z      -0.71374    0.10709  -6.665 2.65e-11 ***
## diff_DD     -0.04884    0.04638  -1.053    0.292
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2145.3  on 3277  degrees of freedom
## Residual deviance: 2064.7  on 3275  degrees of freedom
```

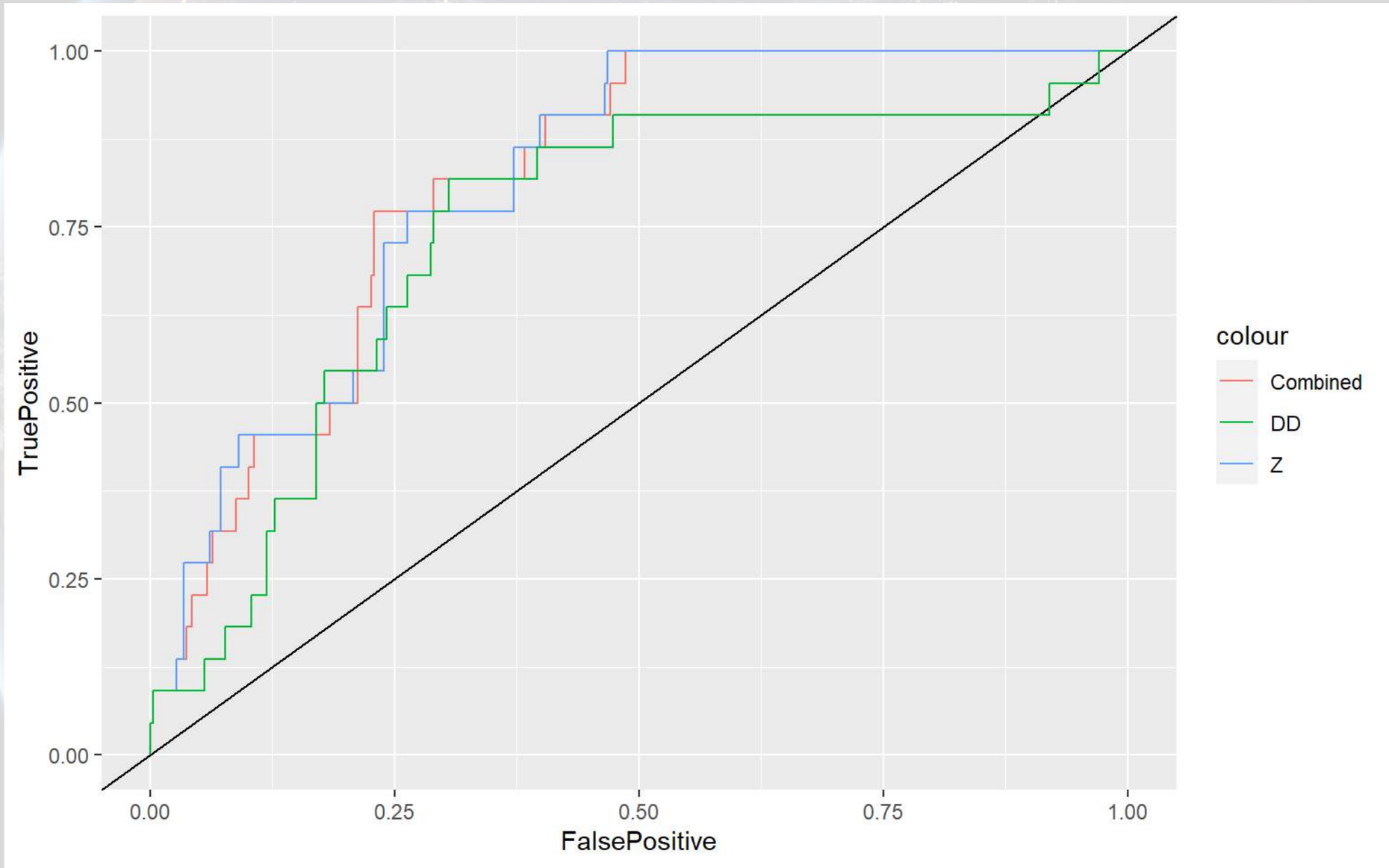
Marginal effects

```
fit_comb %>%
  margins::margins() %>%
  summary()
```

```
##   factor     AME     SE       z      p   lower   upper
##  diff_DD -0.0043 0.0041 -1.0525 0.2926 -0.0122  0.0037
##   diff_Z -0.0625 0.0094 -6.6473 0.0000 -0.0809 -0.0441
```

> The `margins::` syntax allows us to call a function without loading the whole library. There is a conflict in the `predict` functions of `ROCR` and `margins`, so this is safer.

ROC Performance on this task




```
##  Combined         Z        DD
## 0.8151596 0.8134671 0.7420213
```

End matter

For next week

- For next week:
  - Second individual assignment
    - Finish by the end of the day next class session
    - Submit on eLearn
  - Datacamp
    - Practice a bit more to keep up to date
      - Using R more will make it more natural
  - Keep thinking about who you want to work with on the project.

Packages used for these slides

- `kableExtra`
- `knitr`
- `lubridate`
- `magrittr`
- `plotly`
- `revealjs`
- `ROCR`
- `tidyverse`

Custom code

```r
# Calculating a 253 day rolling standard deviation in R
crsp <- crsp %>%
  group_by(gvkey) %>%
  mutate(ret.sd = rollapply(data=ret, width=253, FUN=sd, align="right", fill=NA, na.rm=T)) %>%
  ungroup()
```