香港樹仁大學
HONG KONG SHUE YAN UNIVERSITY

RGC IIDS Accounting Hong Kong Shue Yan | RGC IIDS

APPLICATIONS OF DATA ANALYTICS TO FORENSIC ACCOUNTING

SPEAKER

Dr. Richard Crowley
Singapore Management University

TIME

6 April 2022 (Wednesday)
10:40am - 12:10pm

ZOOM MEETING

This project is fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project no. UGC/IIDS-B01-20)

FREE REGISTRATION

Misreporting Detection: Past to Future

Applications of data analytics to forensic accounting

RGC IIDS
Accounting
Hong Kong Shue Yan

6 April 2022

Dr. Richard M. Crowley
rcrowley@smu.edu.sg
http://rmc.link/

Frontmatter

About me

- Assistant Professor of Accounting at SMU since 2016
- **Research**: Approaching accounting disclosure problems using AI/ML
  - Fraud detection based on annual report content
  - Fine-grained measurement of context within annual reports
  - Corporate and executive social media posting
  - WIP: COVID-19 social media discussion
  - WIP: Impact of fake news legislation
- **Teaching**
  - PhD: Machine Learning for Social Science; Accounting Theory
  - UG: Forecasting and Forensic Analytics; Financial Accounting

> General focus: How non-quantitative information flows in economies and society, with an eye to methodology (especially NLP and ML for econometrics)

Misreporting: A simple definition

> Errors that affect firms' accounting statements or disclosures which were done seemingly *intentionally* by management or other employees at the firm.

> Our focus today is misreporting **detection**

Why do we care?

The 10 most expensive US corporate frauds cost *shareholders* **12.85B USD**

- The above figure is missing:
  - *GDP impacts*: Enron’s collapse cost **~35B USD**
  - *Societal costs*: Lost jobs, lost confidence in the economy and government
  - Any *negative externalities*, e.g. new compliance costs borne by others
  - *Inflation*: In current dollars it is even higher

Catching even 1 major fraud **as they happen** could save billions of dollars

What will we discuss today?

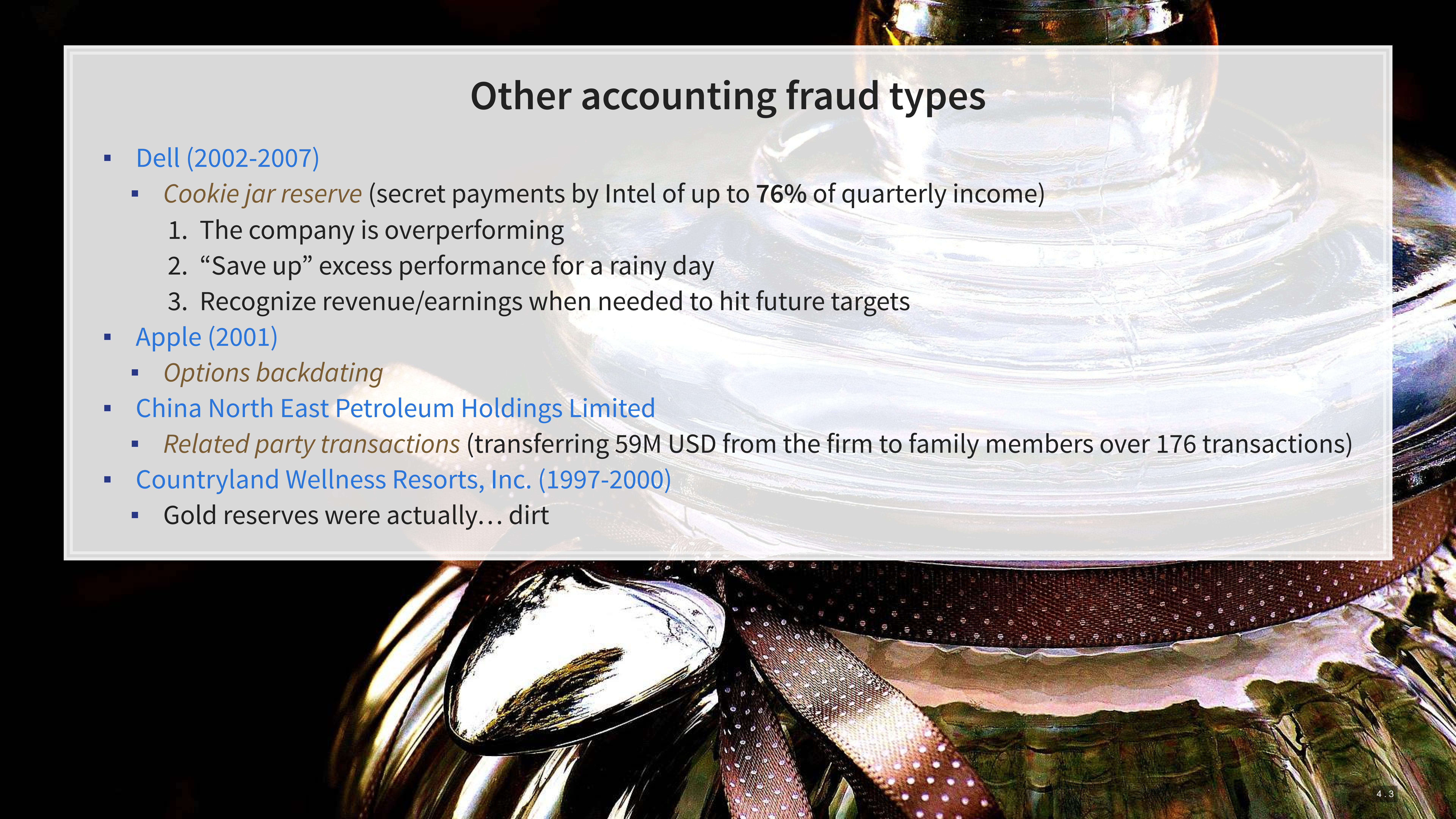
1. What is misreporting, and how is it typically measured?
2. Econometric considerations
3. Older approaches for misreporting detection
   - Financial Ratios
   - F-Score
   - Textual style
4. Newer methods for misreporting detection
   - Machine learned document content as an input into our process
   - Brown, Crowley, and Elliott (2020 JAR)
5. Extending BCE using machine learning regression

Misreporting

Traditional accounting fraud

1. A company is underperforming
2. Someone at the company cooks up some scheme to increase earnings
3. Create accounting statements using the fake information

- Wells Fargo’s opening of accounts without customer’s consent from 2002-2016 is a standard, though extreme, example
  - Led to a $3B USD settlement with the US government



Other accounting fraud types

- Dell (2002-2007)
    - *Cookie jar reserve* (secret payments by Intel of up to **76%** of quarterly income)
        1. The company is overperforming
        2. “Save up” excess performance for a rainy day
        3. Recognize revenue/earnings when needed to hit future targets
- Apple (2001)
    - *Options backdating*
- China North East Petroleum Holdings Limited
    - *Related party transactions* (transferring 59M USD from the firm to family members over 176 transactions)
- Countryland Wellness Resorts, Inc. (1997-2000)
    - Gold reserves were actually… dirt

What misreporting measures are there? (US)

1. US SEC AAERs: **Accounting and Auditing Enforcement Releases**
   - Highlight larger/more important cases, written by the SEC
   - Most of our discussion today will focus on this data source
2. 10-K/A filings (“10-K” ⇒ annual report, “/A” ⇒ amendment)
   - Note: not all 10-K/A filings are caused by fraud!
     - Benign corrections or adjustments can also be filed as a 10-K/A
     - Note: Audit Analytics’ write-up on this for 2017
3. By the US government through a 13(b) action
4. In a note inside a 10-K filing
   - These are sometimes referred to as “little r” restatements
5. In a press release, which is later filed with the US SEC as an 8-K
   - 8-Ks are filed for many other reasons too though
6. A whole host of other classifications proposed by data vendors (Audit Analytics) and research papers

Where are we at?

> Misreporting happens in many ways, for many reasons

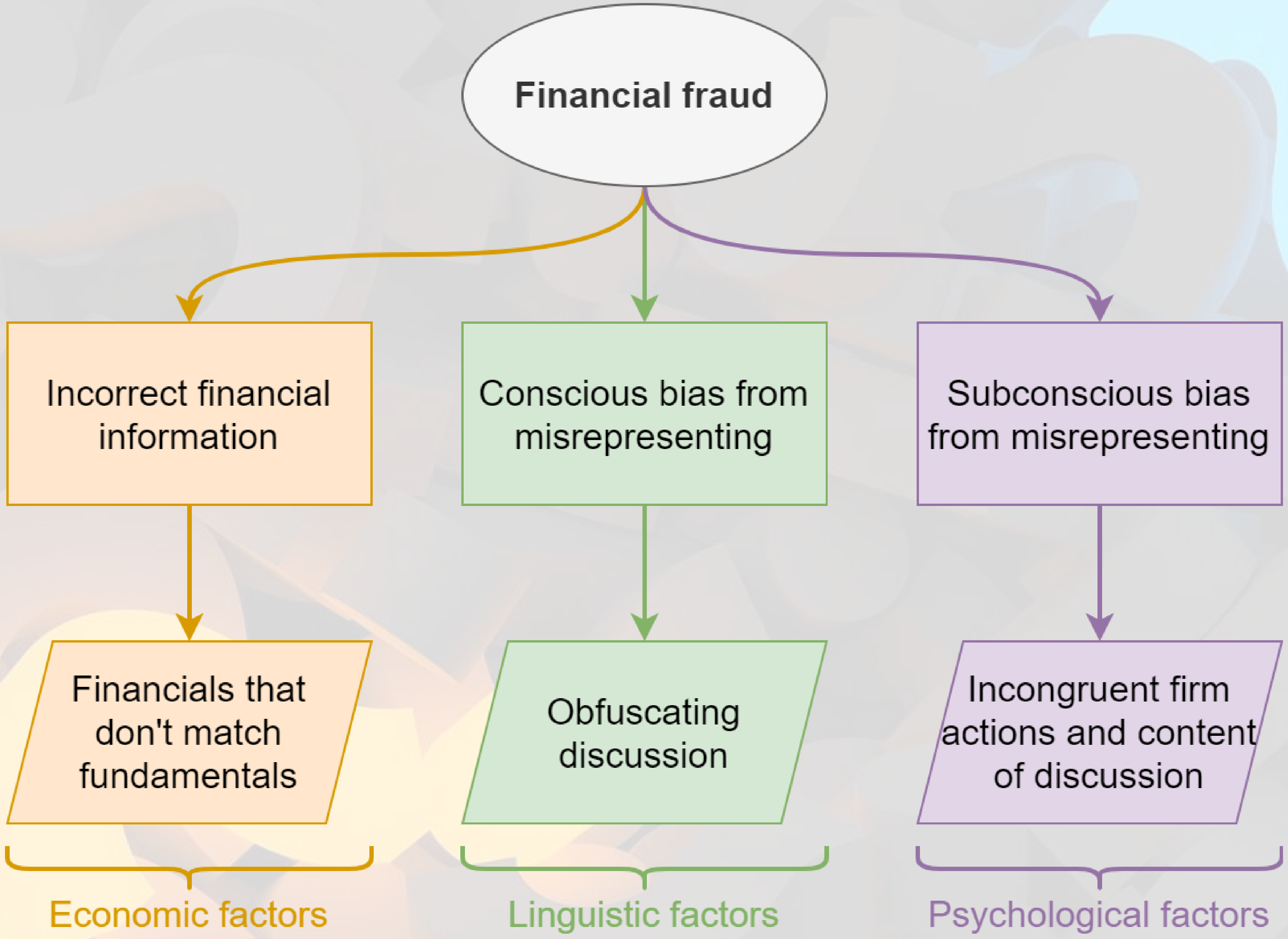
- All of them are important to capture
- All of them affect accounting numbers differently
- None of the individual methods are frequent…

> It is disclosed in many places. All have subtly different meanings and implications

- We need to be careful here (or check multiple sources)

> This is a hard problem!

Through what lenses can we view misreporting?



Financial fraud
Incorrect financial information
Conscious bias from misrepresenting
Subconscious bias from misrepresenting
Financials that don't match fundamentals
Obfuscating discussion
Incongruent firm actions and content of discussion
Economic factors
Linguistic factors
Psychological factors

Econometric considerations

Training and Testing

- Training is done using 5-year windows
- Testing is the year after the training window

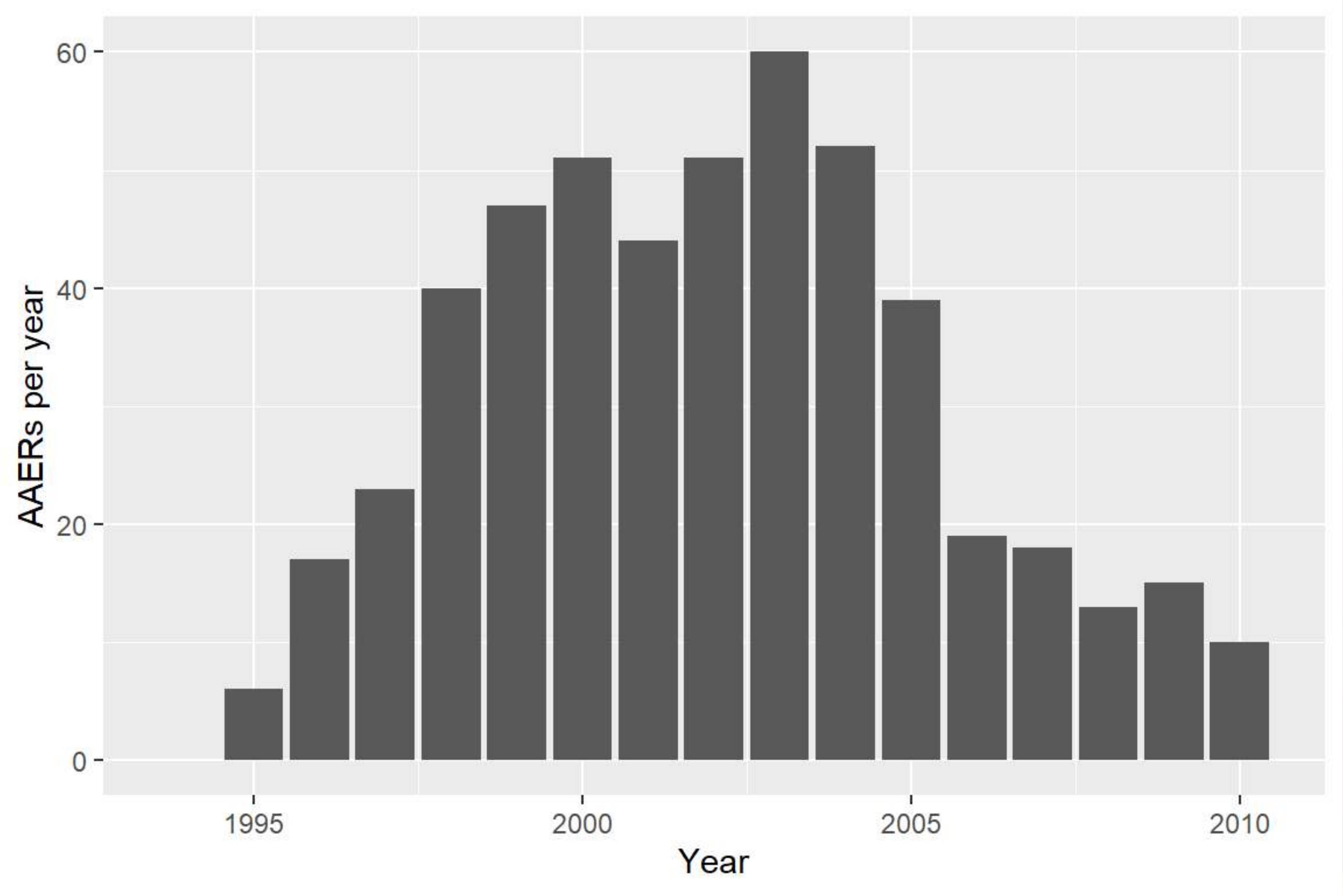
> There is a big issue in constructing training samples

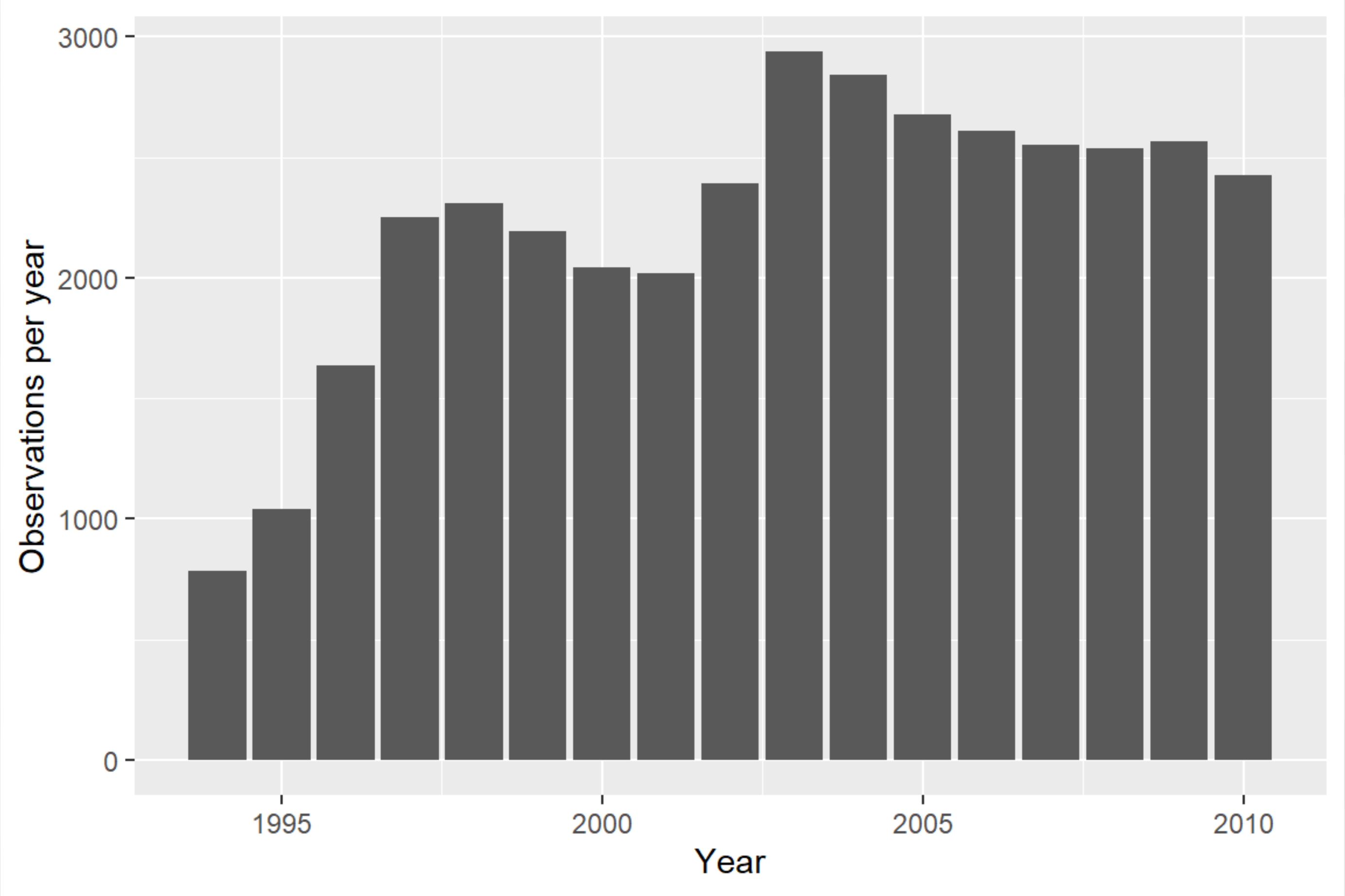
- There is a significant lag between when a fraud is caught and when a fraud actually happened!
  - E.g., to mirror the available information in early 2004, we should *censor* AAERs in the training data such that we only capture AAERs known by the end of 2003

year	year_found	aaer	aaer_as_of_2004
1999	2001	1	1
2001	2003	1	1
2003	2006	1	0

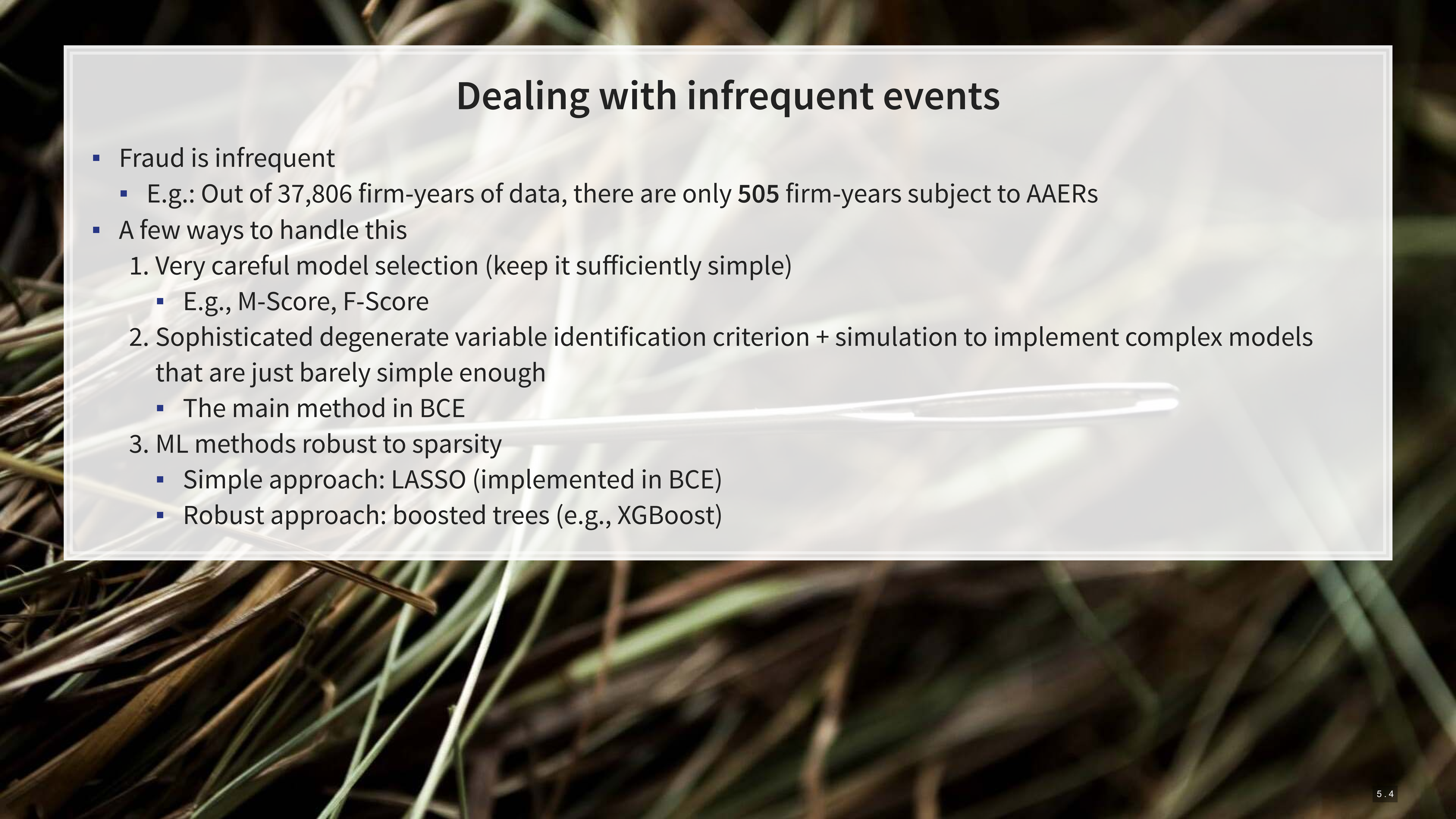
> Not an issue with testing data: Testing should emulate where we want to make an optimal choice in real life

Event frequency





Very low event frequencies can make things tricky



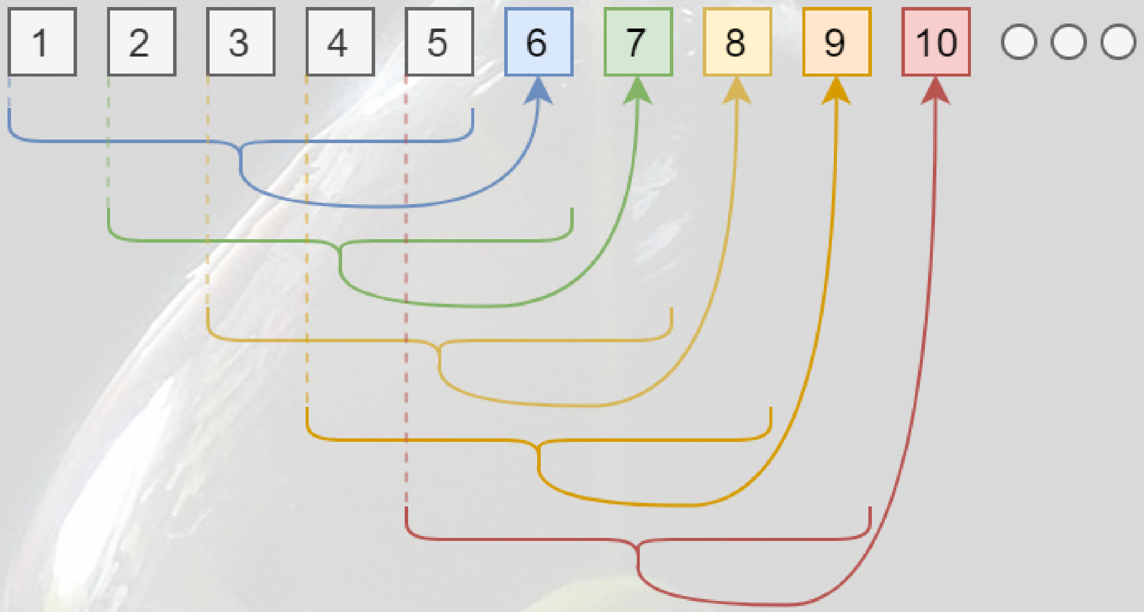
Dealing with infrequent events

- Fraud is infrequent
  - E.g.: Out of 37,806 firm-years of data, there are only **505** firm-years subject to AAERs
- A few ways to handle this
  1. Very careful model selection (keep it sufficiently simple)
     - E.g., M-Score, F-Score
  2. Sophisticated degenerate variable identification criterion + simulation to implement complex models that are just barely simple enough
     - The main method in BCE
  3. ML methods robust to sparsity
     - Simple approach: LASSO (implemented in BCE)
     - Robust approach: boosted trees (e.g., XGBoost)

Moving target

Misreporting evolves over time

- Implement a moving window approach
  - 5 years for training + 1 year for testing
  - We use data from 1994 through 2012: 14 possible windows
- Ex.: to predict misreporting in 2010, train on data from 2005 to 2009



Problem: Now we have 14 models…

Logistic iteration

1. Fit a logit using a Newton-Raphson solver for 50 iterations
   - Note: This presentation switched to IWLS for fitting
2. Check convergence for signs of quasi-completeness
   - Standard errors will be in the millions if quasi-complete
   - If quasi-complete, drop the next *least independent* measure and restart
3. Run a 500 iteration logit using a Newton-Raphson solver
   - Note: Not needed under IWLS
4. Recheck convergence
   - If failed, drop the next *least independent* measure and restart

> We will essentially get the most complex feasible model with the most independent set of features

Identifying the *least independent* measure

> Use a QR decomposition

- Lets us determine an order for dropping inputs
- $X = Q \times R$, where X is our feature matrix, Q is an orthogonal matrix, and R is the transformation
  - More weight on the diagonal element in R means more independent (effectively)
  - Same underlying method as a Gram-Schmidt process

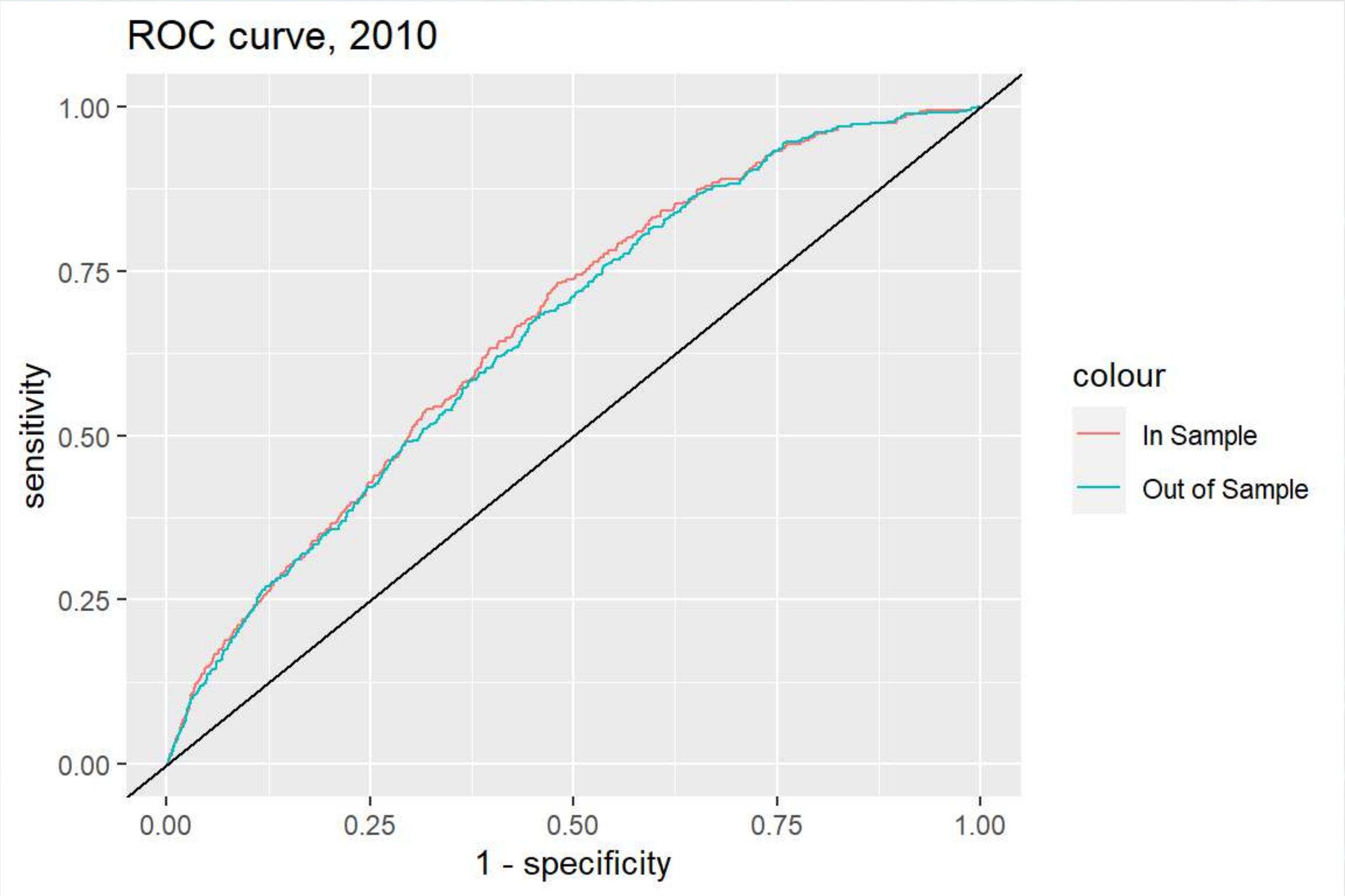
> Independentness is a useful criterion for removing features with lower likelihood of being useful

```r
independentness <- function(X) {
  which.min(abs(diag(qr(X)$qr)))
}
```

```python
import numpy as np
def independentness(X):
    Q, R = np.linalg.qr(X)
    D = np.abs(R.diagonal())
    return np.argmin(D)
```

Model performance: ROC AUC

- **R**eceiver **O**perator **C**urve
  - ROC curve compares sensitivity and specificity of a model
    - Sensitivity: True positive rate
    - Specificity: True negative rate
- A better measure has a curve closer to the upper left corner
- A random measure has a curve at a 45-degree line



> **A**rea **U**nder the **C**urve (AUC): What is the probability that a randomly selected `AAER=1` is ranked higher than a randomly selected `AAER=0`? A good score is above 0.70

Traditional approaches

Financial Ratios

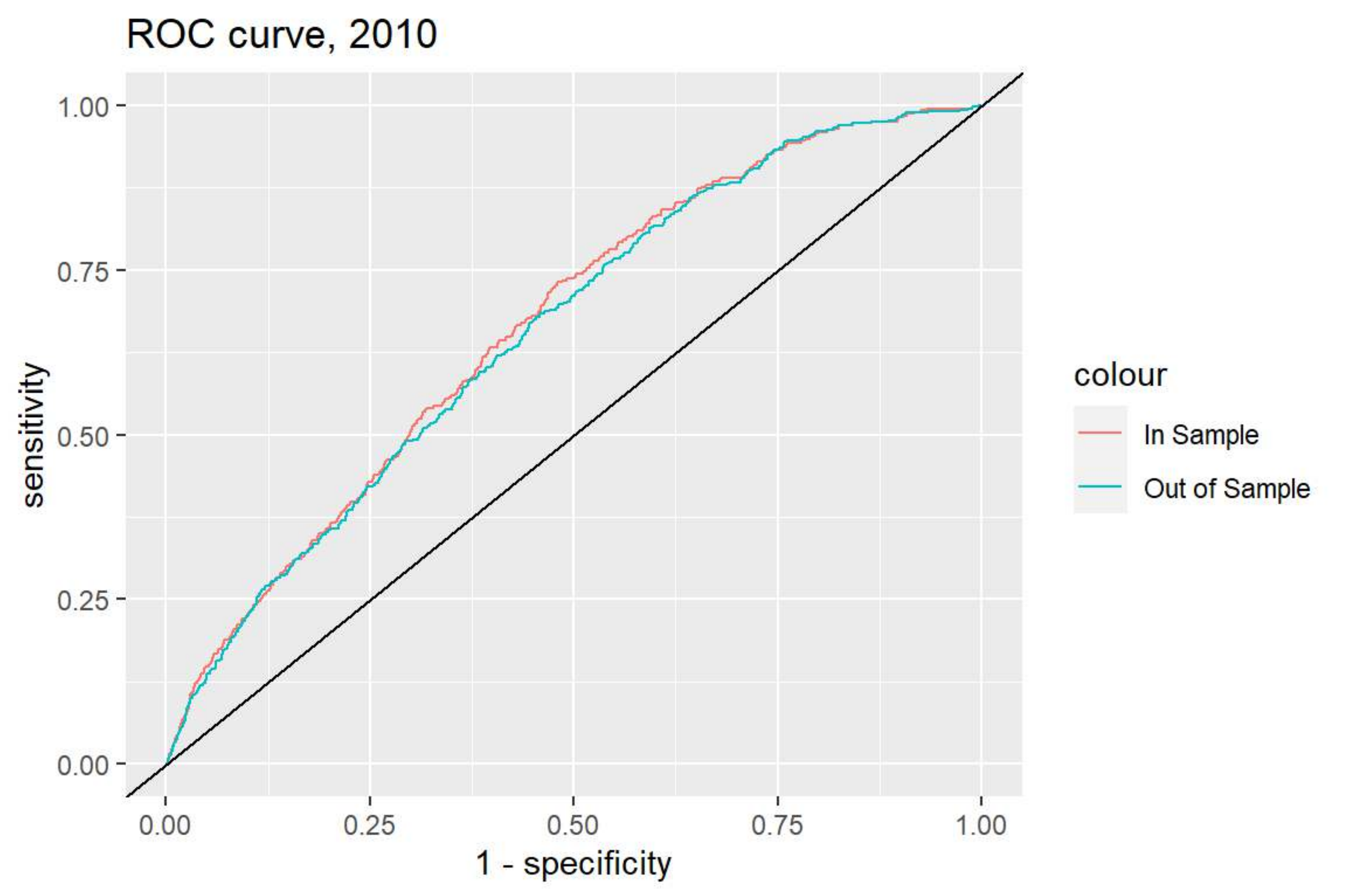
- Many financial measures and ratios can help to predict fraud

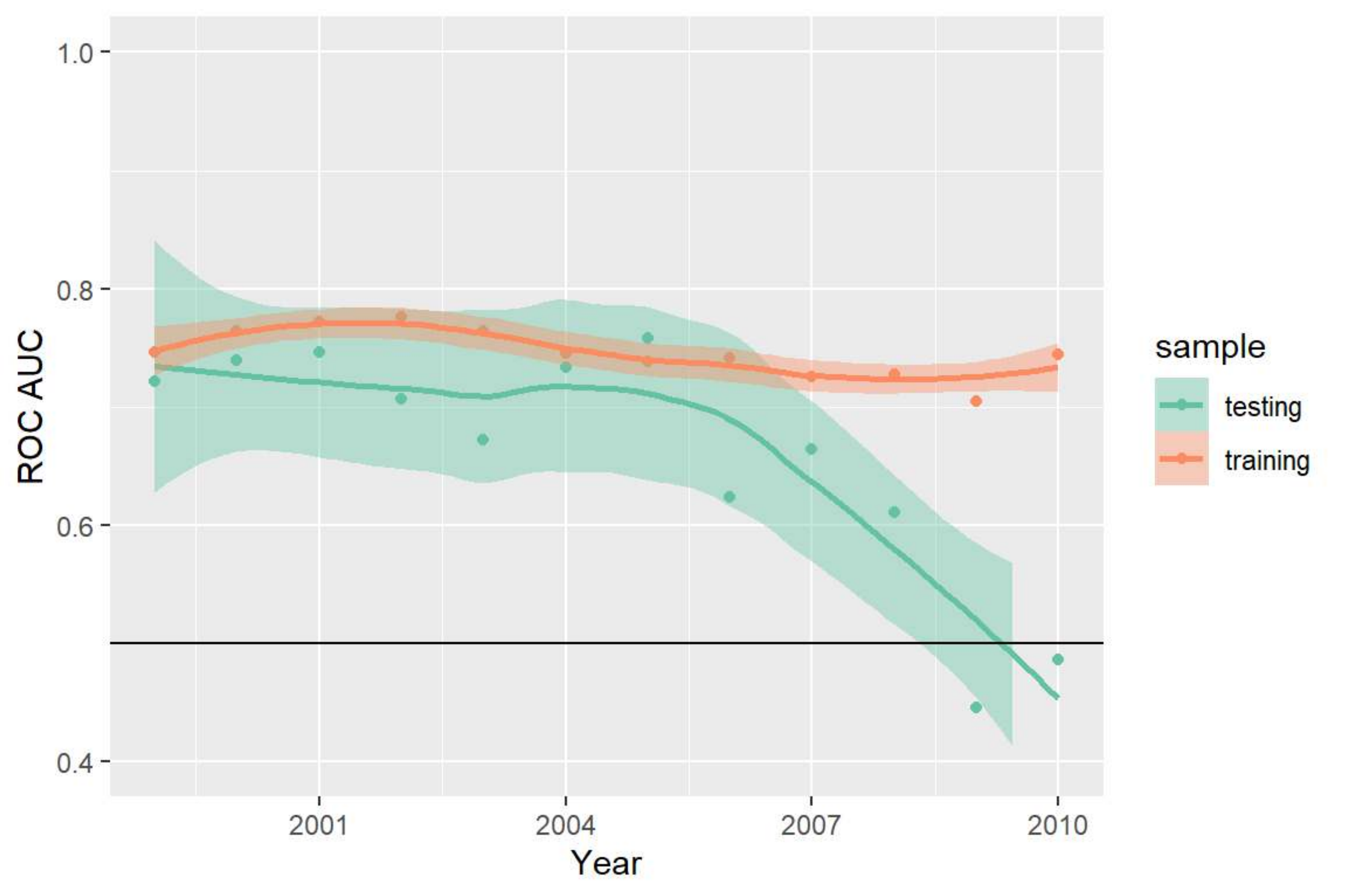
- EBIT
- Earnings / revenue
- ROA
- Log of liabilities
- liabilities / equity
- liabilities / assets
- quick ratio
- Working capital / assets
- Inventory / revenue
- inventory / assets
- earnings / PP&E
- A/R / revenue

- Change in revenue
- Change in A/R + 1
- $> 10\%$ change in A/R
- Change in gross profit + 1
- $> 10\%$ change in gross profit
- Gross profit / assets
- Revenue minus gross profit
- Cash / assets
- Log of assets
- PP&E / assets
- Working capital

> Purely economic: Misreporting firms’ financials should be different than expected

Performance: Financial ratios





Good performance early on, but poor performance 2006 onward

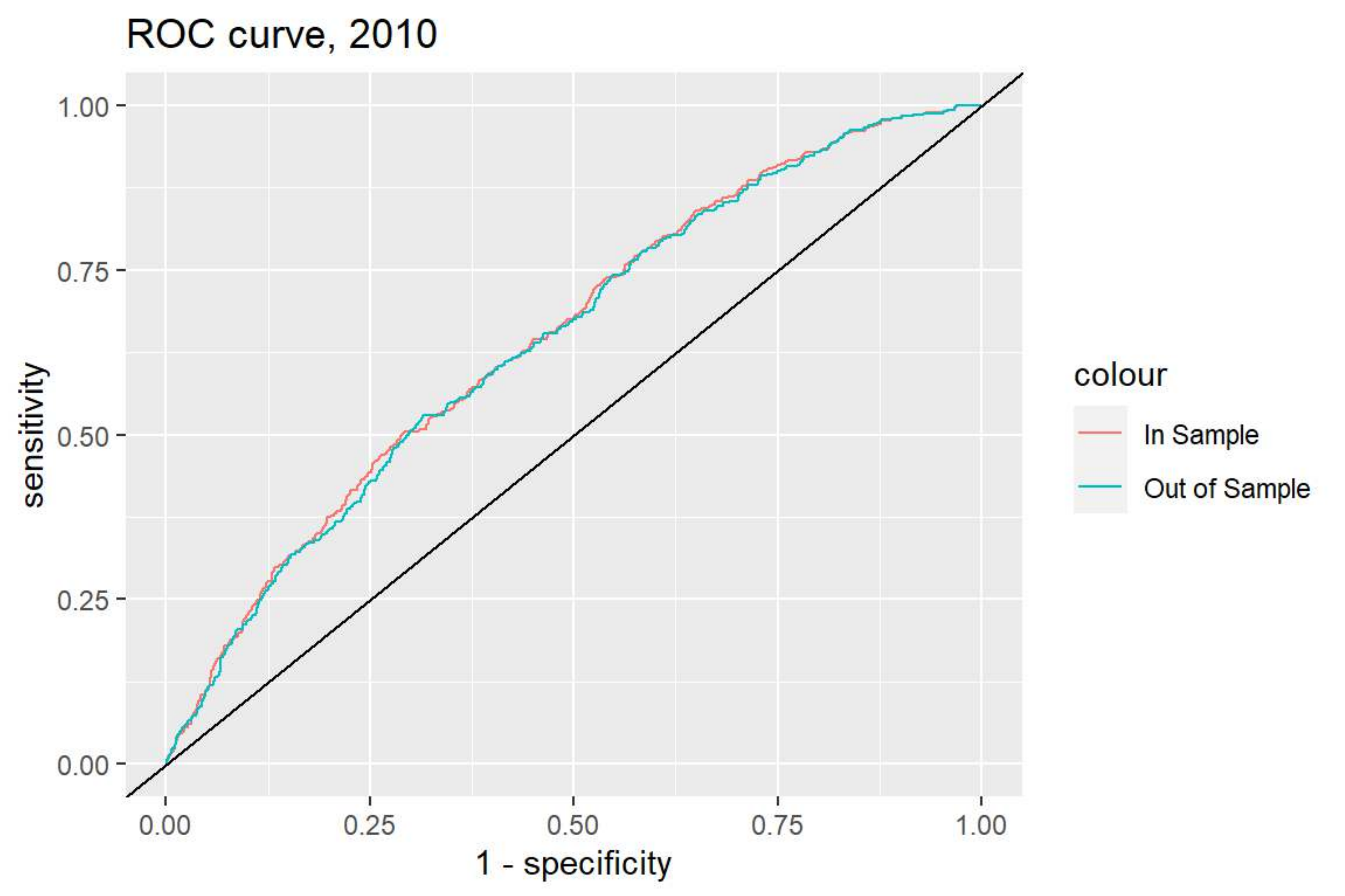
F-Score

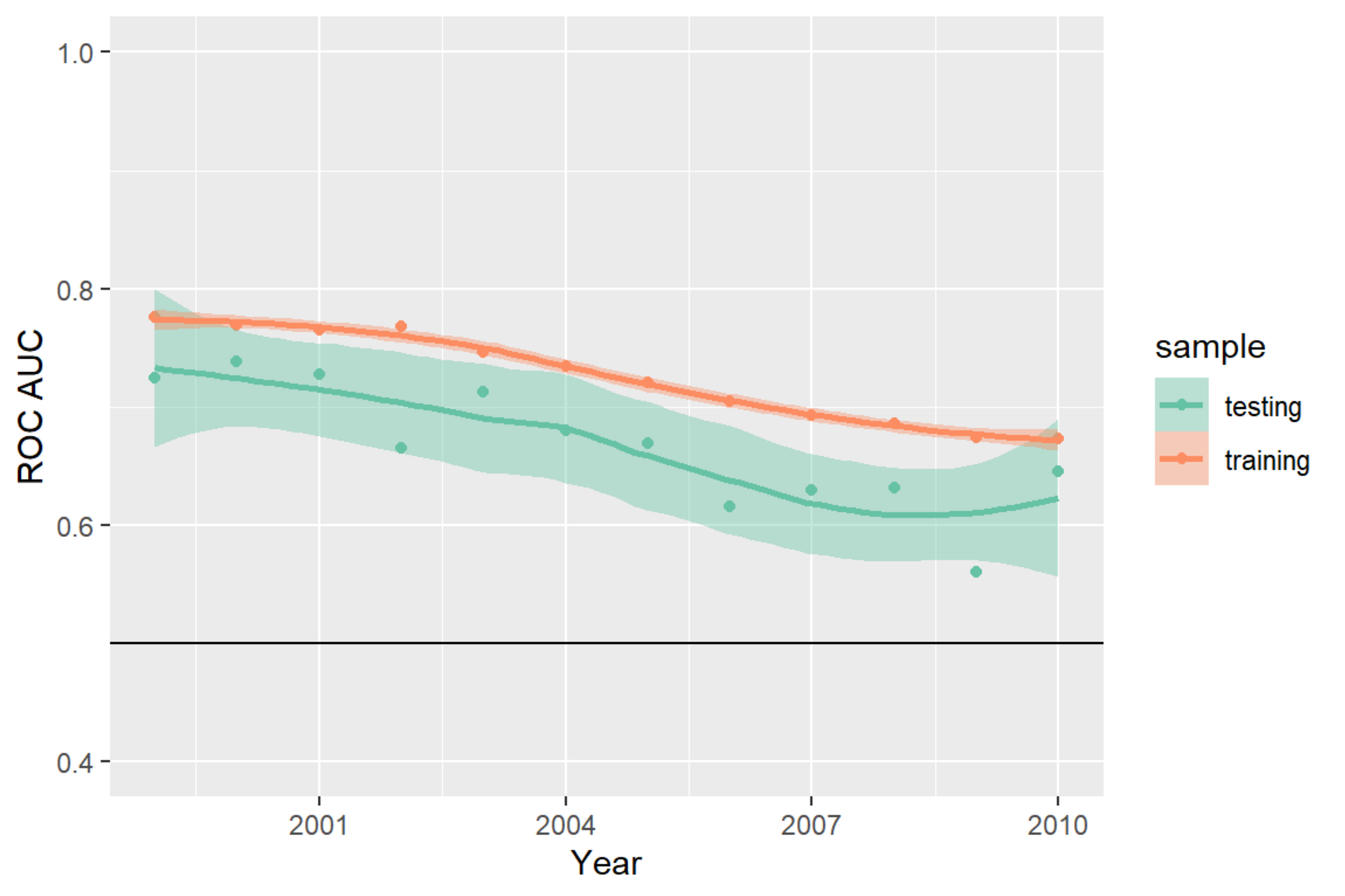
- Log of assets
- Total accruals
- % change in A/R
- % change in inventory
- % soft assets
- % change in sales from cash
- % change in ROA
- Indicator for stock/bond issuance
- Indicator for operating leases
- BV equity / MV equity

- Lag of stock return minus value weighted market return
- **Below are BCE's additions**
- Indicator for mergers
- Indicator for Big N auditor
- Indicator for medium size auditor
- Total financing raised
- Net amount of new capital raised
- Indicator for restructuring

> Based on Dechow, Ge, Larson and Sloan (2011)

Performance: F-Score





Middling in the mid-2000s, better in later years, still dropping

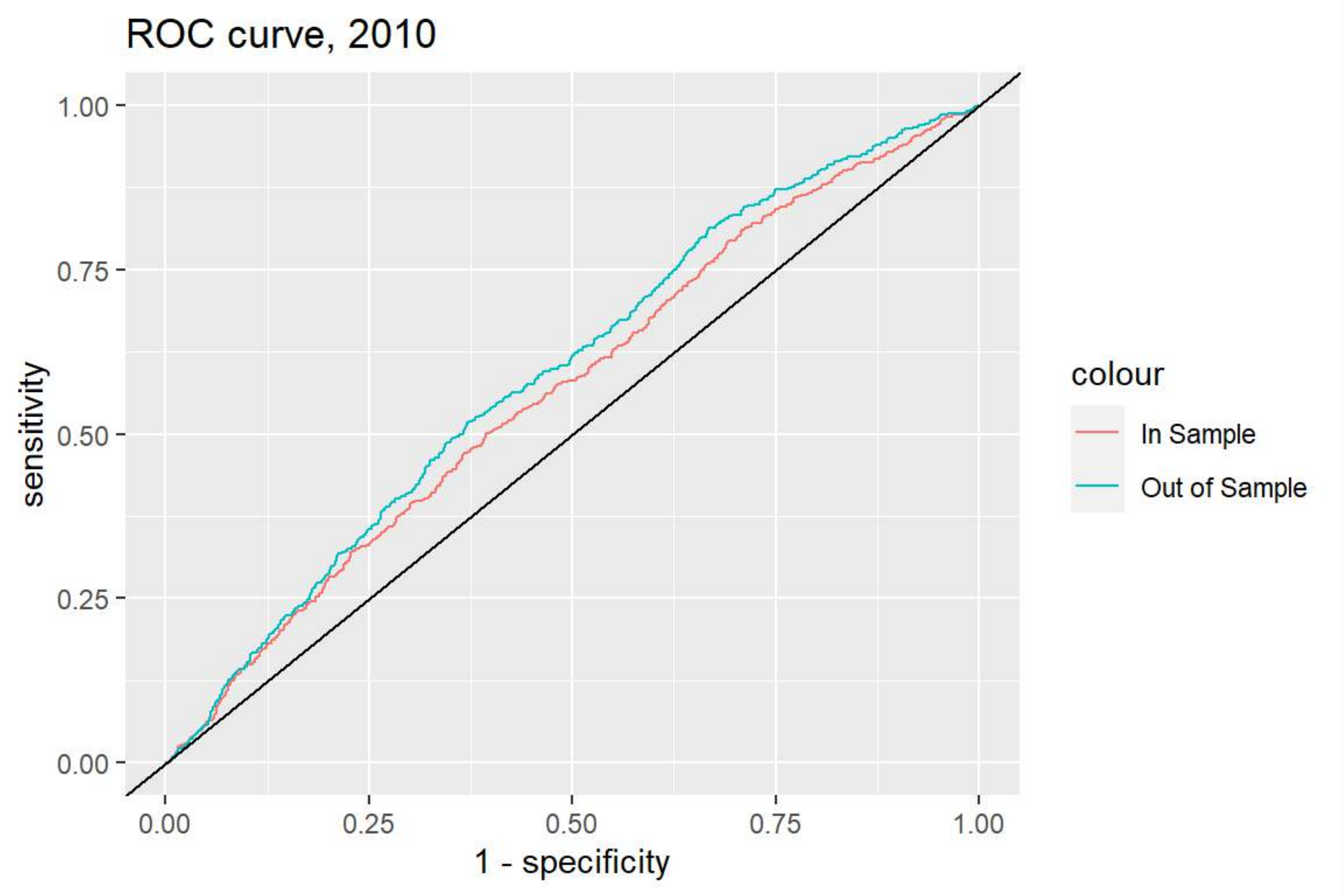
Style

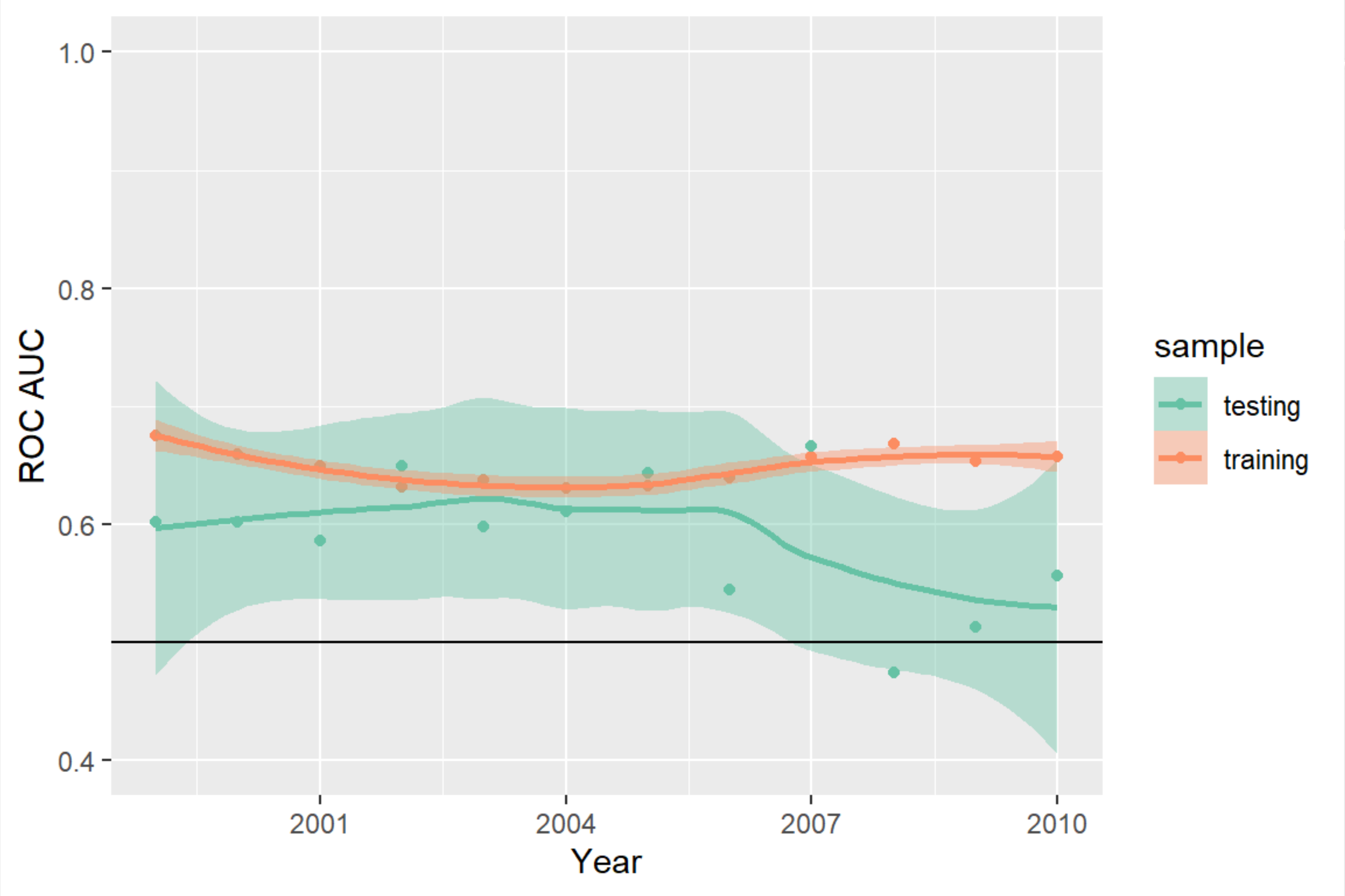
- Log of # of bullet points + 1
- # of characters in file header
- # of excess newlines
- Amount of html tags
- Length of cleaned file, characters
- Mean sentence length, words
- S.D. of word length
- S.D. of paragraph length (sentences)

- Word choice variation
- Readability
  - Coleman Liau Index
  - Fog Index
- % active voice sentences
- % passive voice sentences
- # of all cap words
- # of !
- # of ?

> Companies that are misreporting probably write their annual report differently

Performance: Style





Pretty poor performance in general, not as severe of a drop in 2010

Idea: Combine F-Score and Style

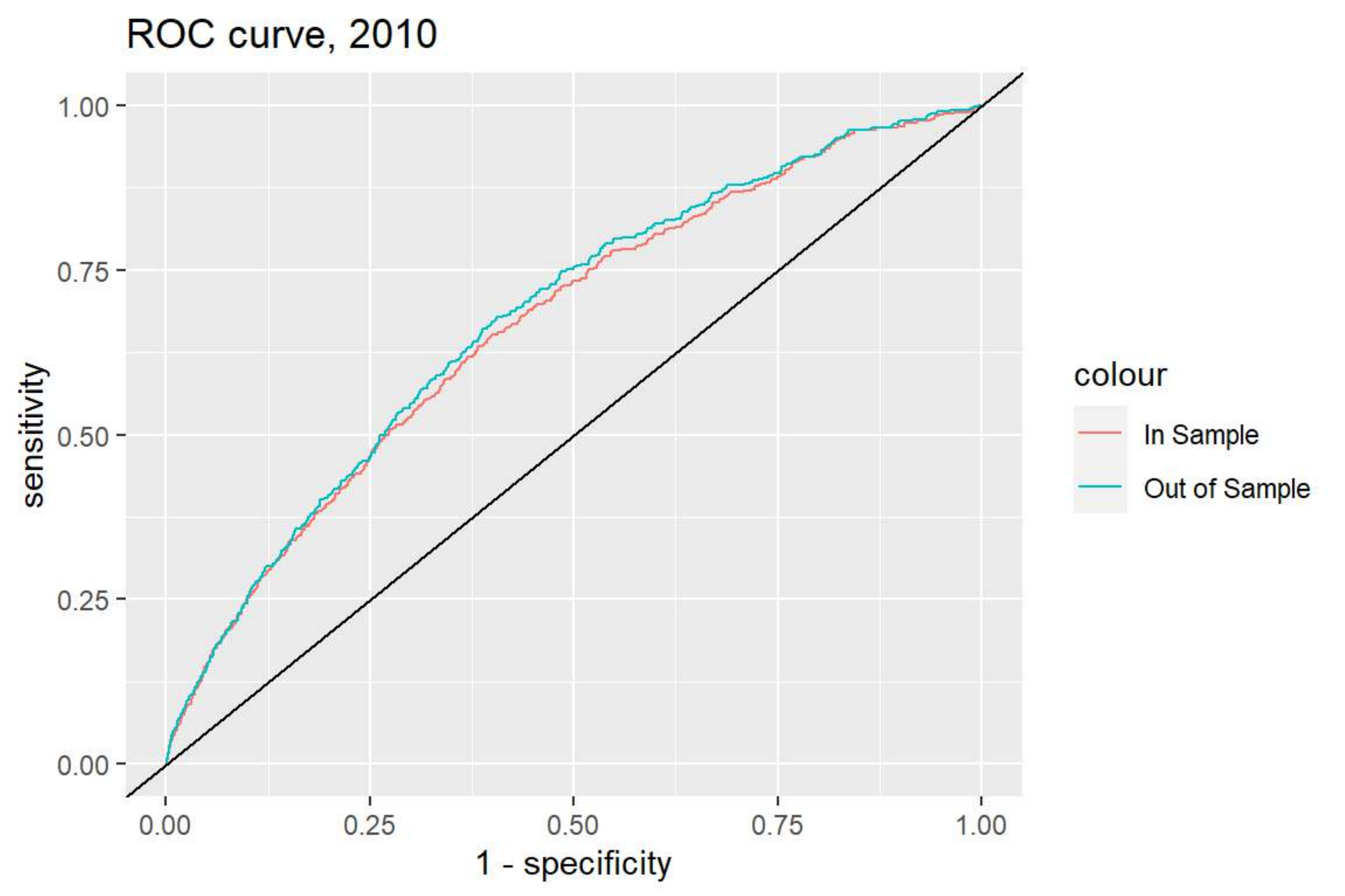
Why is it appropriate to combine the F-Score and Style models?

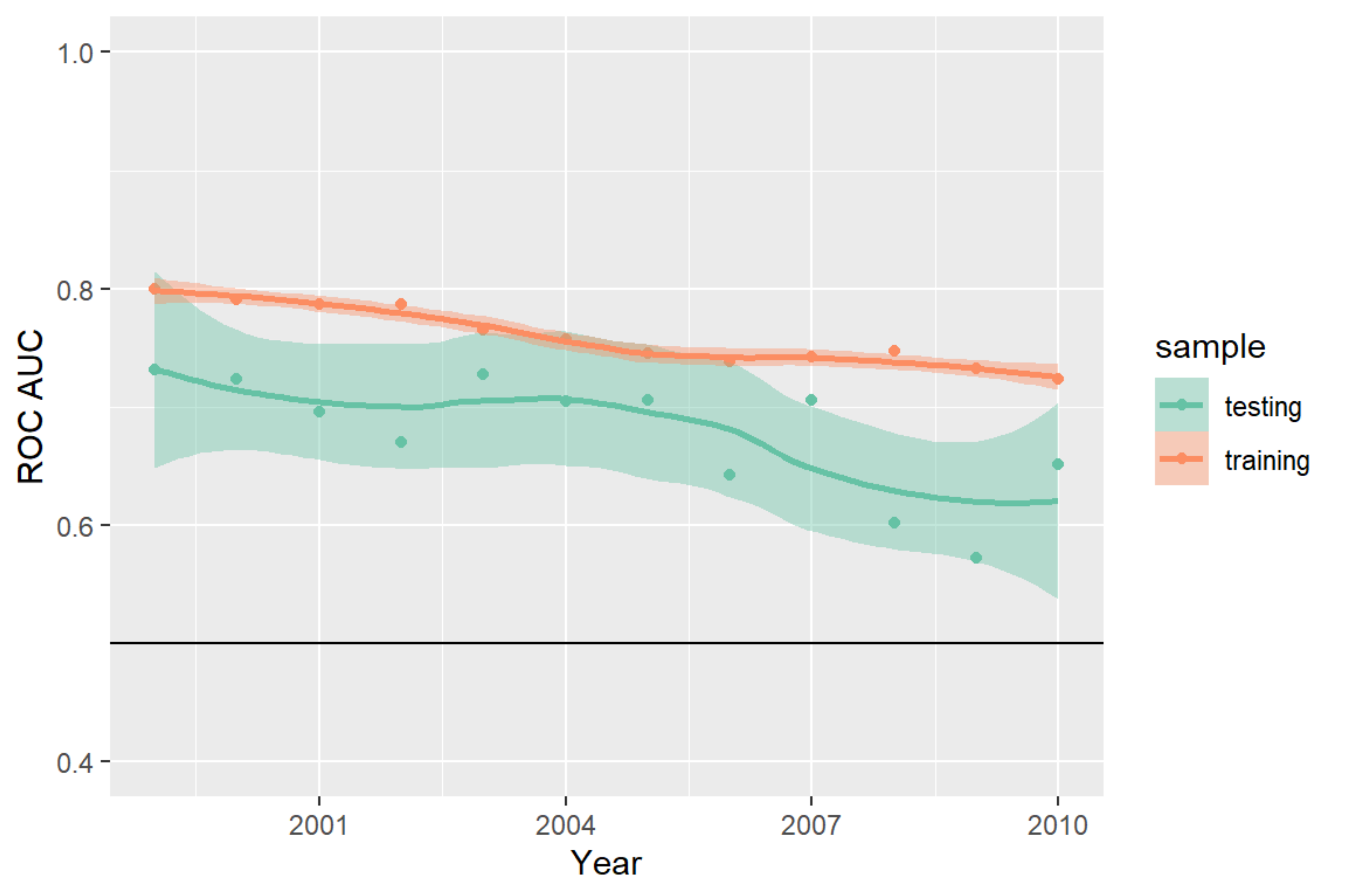
- F-Score: Parsimonious financial model
- Style: Textual characteristics

Little theoretical overlap

Limited multicollinearity across measures

Performance: F-Score + Style





A bit better! F-Score and Style are complementary (empircally)

The BCE model

The BCE approach

- Retain the variables from the other regressions
- Add in a machine-learning based measure quantifying how much documents talked about different topics common across all filings
  - Learn on the training sample windows

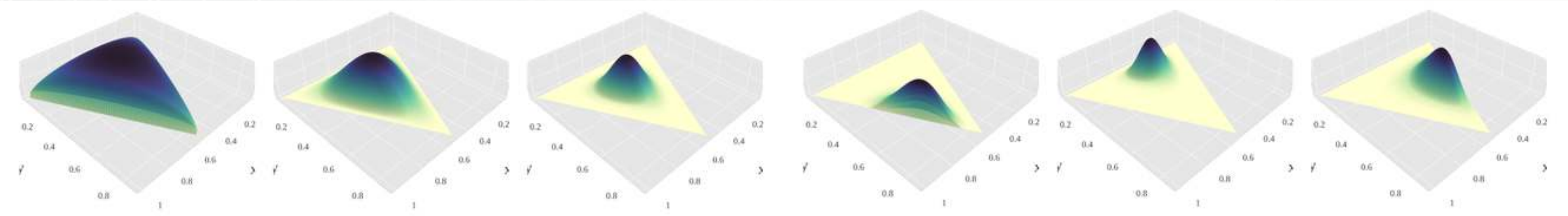
> Why use document content?

- From communications and psychology:
  - When people are trying to deceive others, what they say is carefully picked
    - Topics chosen are intentional
- Putting this in a business context:
  - If you are manipulating inventory, you don't talk about it

> How to measure document content? Latent Dirichlet Allocation (LDA)

What is LDA?

- **L**atent **D**irichlet **A**llocation
- One of the most popular methods under the field of *topic modeling*
- LDA is a Bayesian method of assessing the content of a document
- LDA is also an unsupervised machine learning approach
  - We don't need to know the topics to get a classification!
- LDA assumes there are a set of topics in each document, and that this set follows a *Dirichlet* prior for each document
  - Words within topics also have a *Dirichlet* prior



More details from the creator

How does it work?

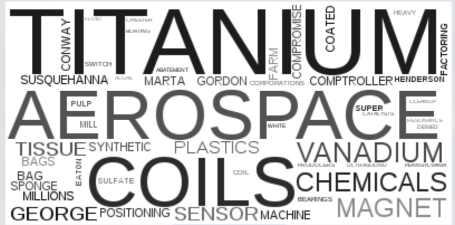
1. LDA reads all the documents
   - Calculates counts of each word within the document, tied to a specific ID used across all documents
2. Uses variation in words within and across documents to infer topics
   - By using a Gibbs sampler to simulate the underlying distributions
     - An MCMC method
     - It boils down to a system where generating a document follows a couple rules:
       1. Topics in a document follow a multinomial/categorical distribution
       2. Words in a topic follow a multinomial/categorical distribution
     - Use words' covariance within and across documents to back out topics in a Bayesian manner

> Caveat: Need to specify the number of topics *ex ante*

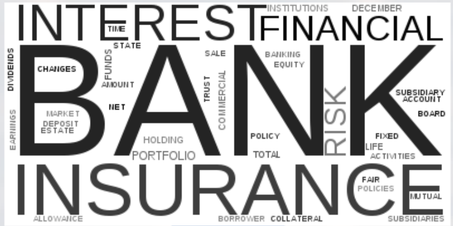
- We determined there to be 31 topics in our setting via simulation

> To run it: BCE used `onlineldavb`, `Gensim` is great for python, and `STM` is great for R

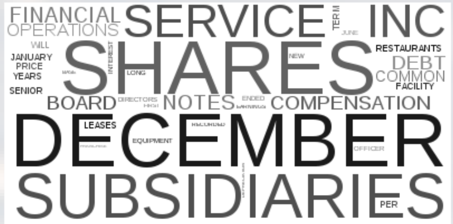
What the topics look like



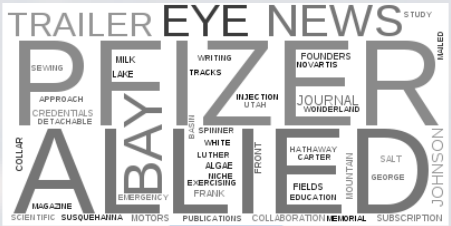
Topic 6



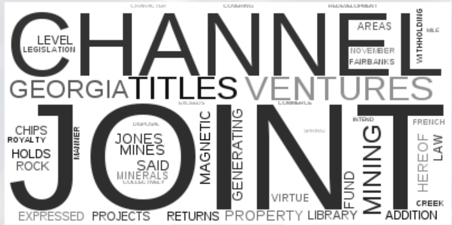
Topic 11



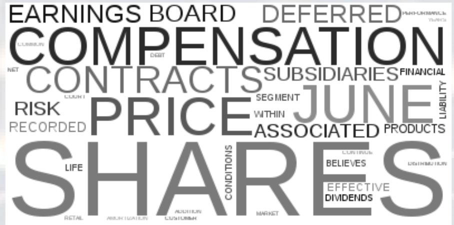
Topic 21



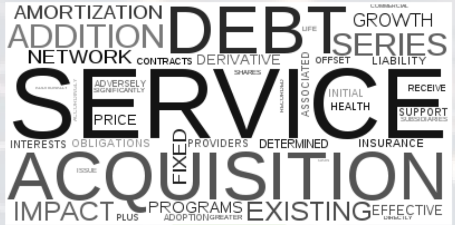
Topic 30



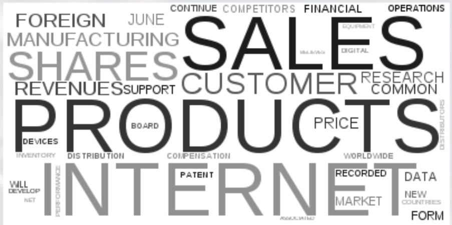
Topic 2



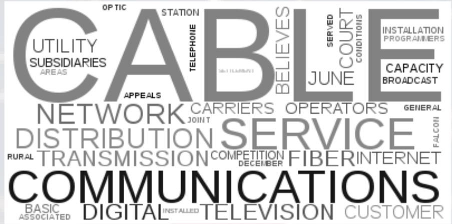
Topic 9



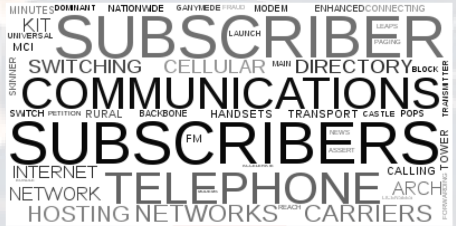
Topic 12



Topic 26



Topic 8



Topic 19

An interactive illustration of a 10 topic model

Implementation details

> The usual addage that data cleaning takes the longest still holds true

1. Annual reports are a mess
   - Fixed width text files; proper html; html exported from MS Word…
   - Embedded hex images
   - Solution: Regexes, regexes, regexes
     - Detailed in the paper's web appendix
2. Stemming, tokenizing, stopwords
3. Feed to LDA
4. Tune hyperparameters (# of topics is most crucial)
   - Tune this by maximizing in-sample prediction ability
5. Finally implement the model
6. Normalize topics by document to a percentage between 0 and 1
7. Separate out industry components of discussion (orthogonalize)

$$topic_{i,firm} = \alpha + \sum_j \beta_{i,j} Industry_{j,firm} + \varepsilon_{i,firm}$$

Experimental validation

Instrument: A word intrusion task

- Which word doesn't belong?
  1. Commodity, Bank, Gold, Mining
  2. Aircraft, Pharmaceutical, Drug, Manufacturing
  3. Collateral, Iowa, Residential, Adjustable

Participants

- 100 individuals on Amazon Turk (20 questions each)
  - Human but not specialized

Quasi-experimental validation

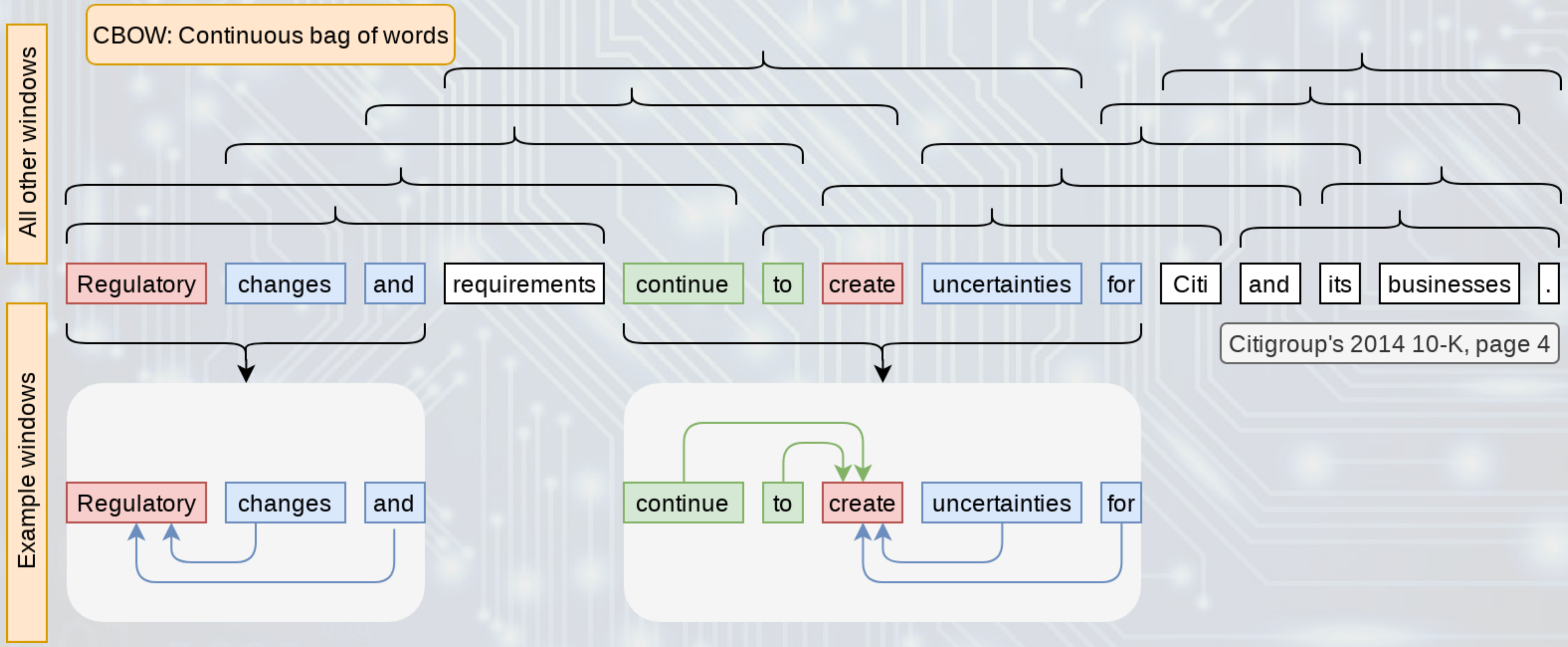
- 3 Computer algorithms (>10M questions each)
  - Not human but specialized
  1. GloVe on general website content
     - Less specific but more broad
  2. Word2vec trained on Wall Street Journal articles
     - More specific, business oriented
  3. Word2vec directly on annual reports
     - Most specific

> These learn the “meaning” of words in a given context

> Run the *exact same* experiment as on humans

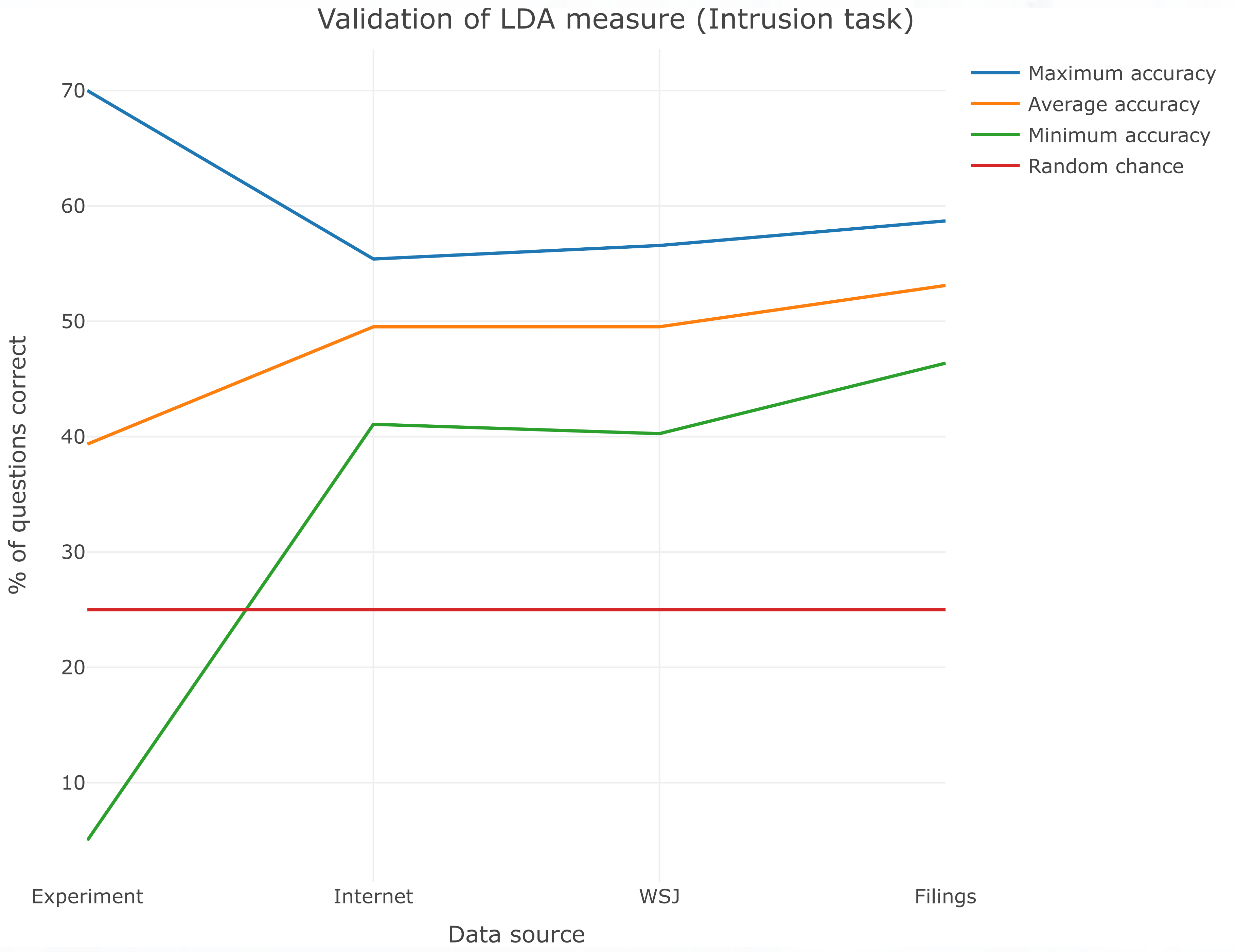
How does word2vec work?

Infer a word’s meaning from the words around it

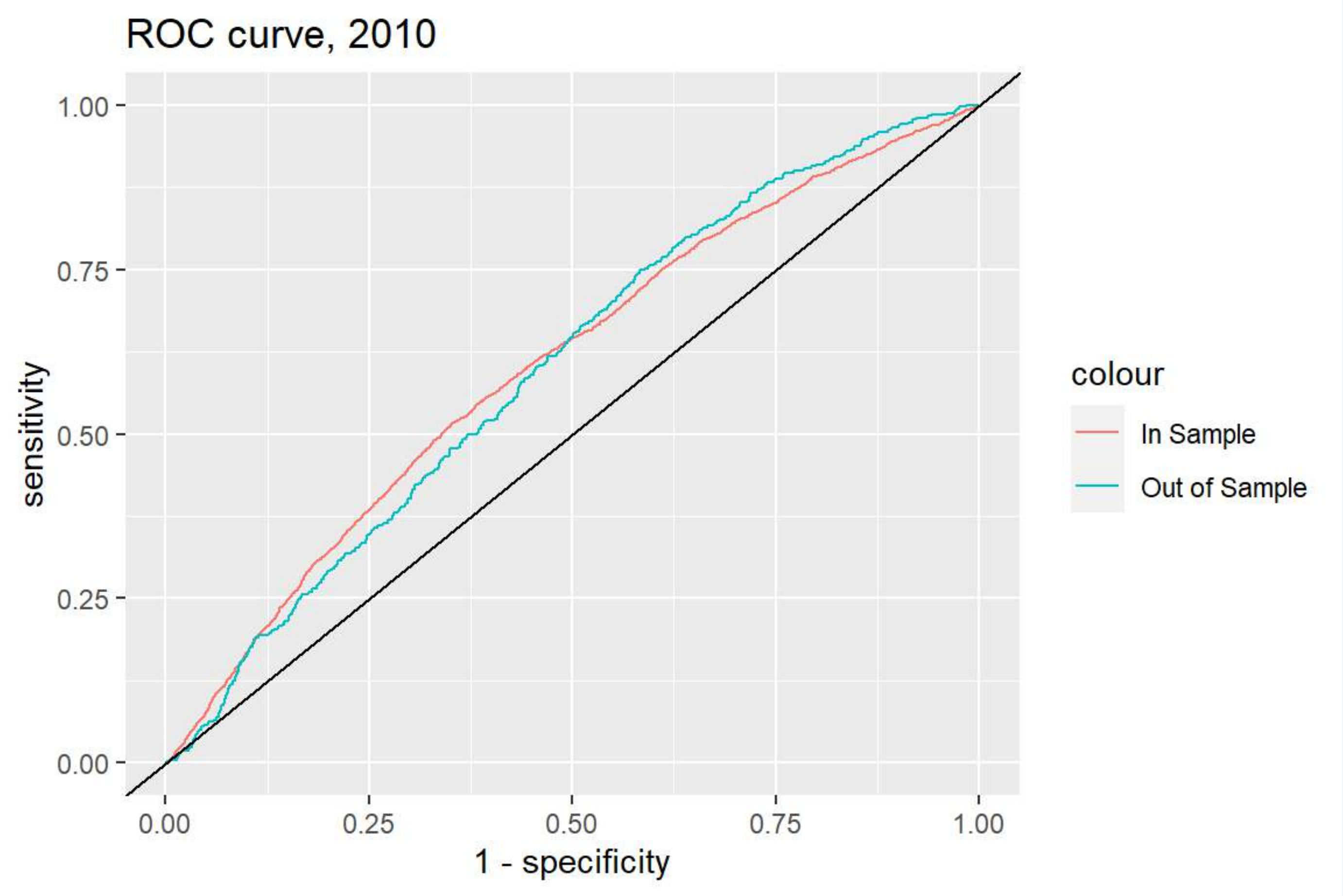


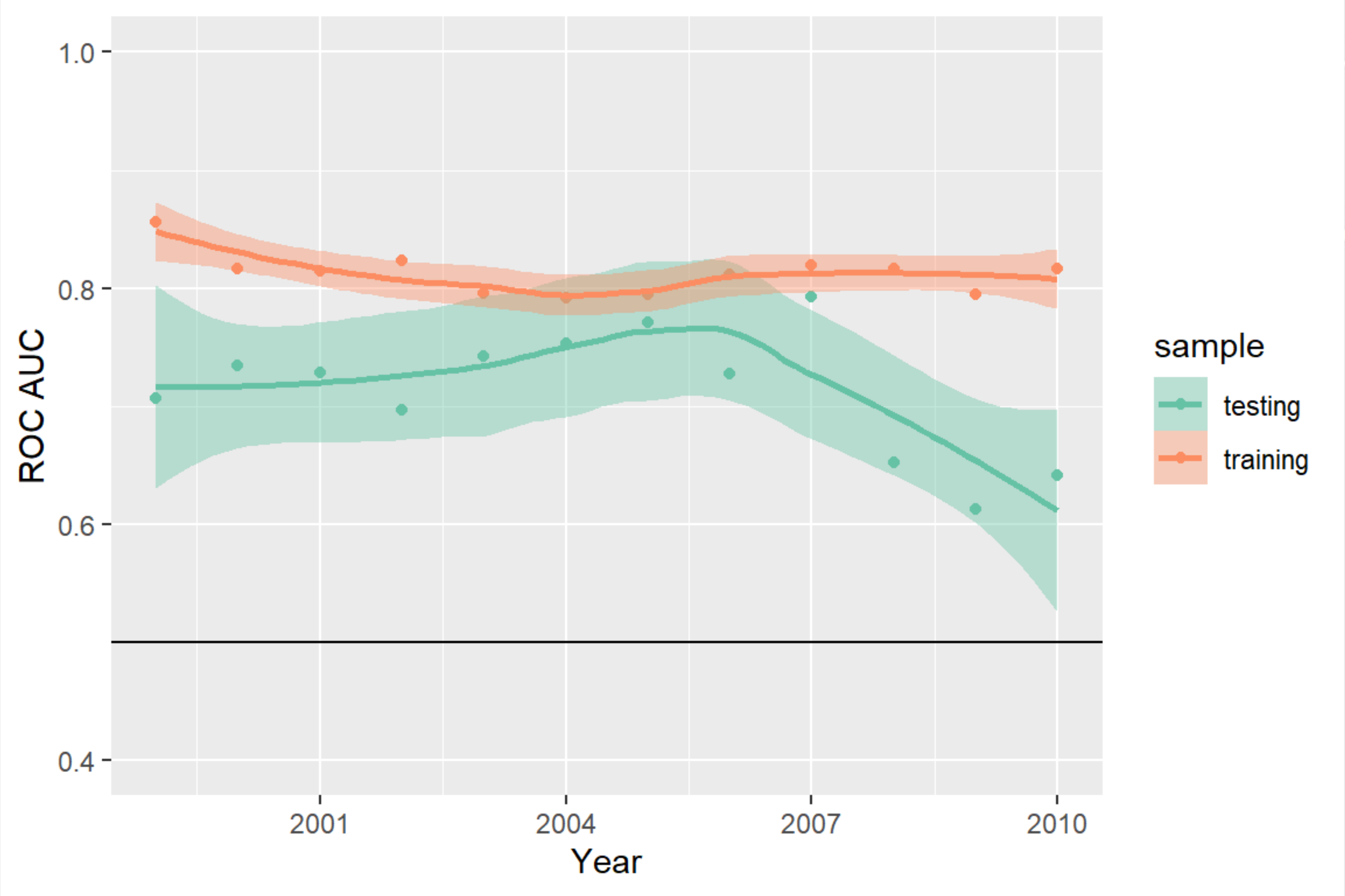
Refered to as CBOW (continuous bag of words)

Experimental results



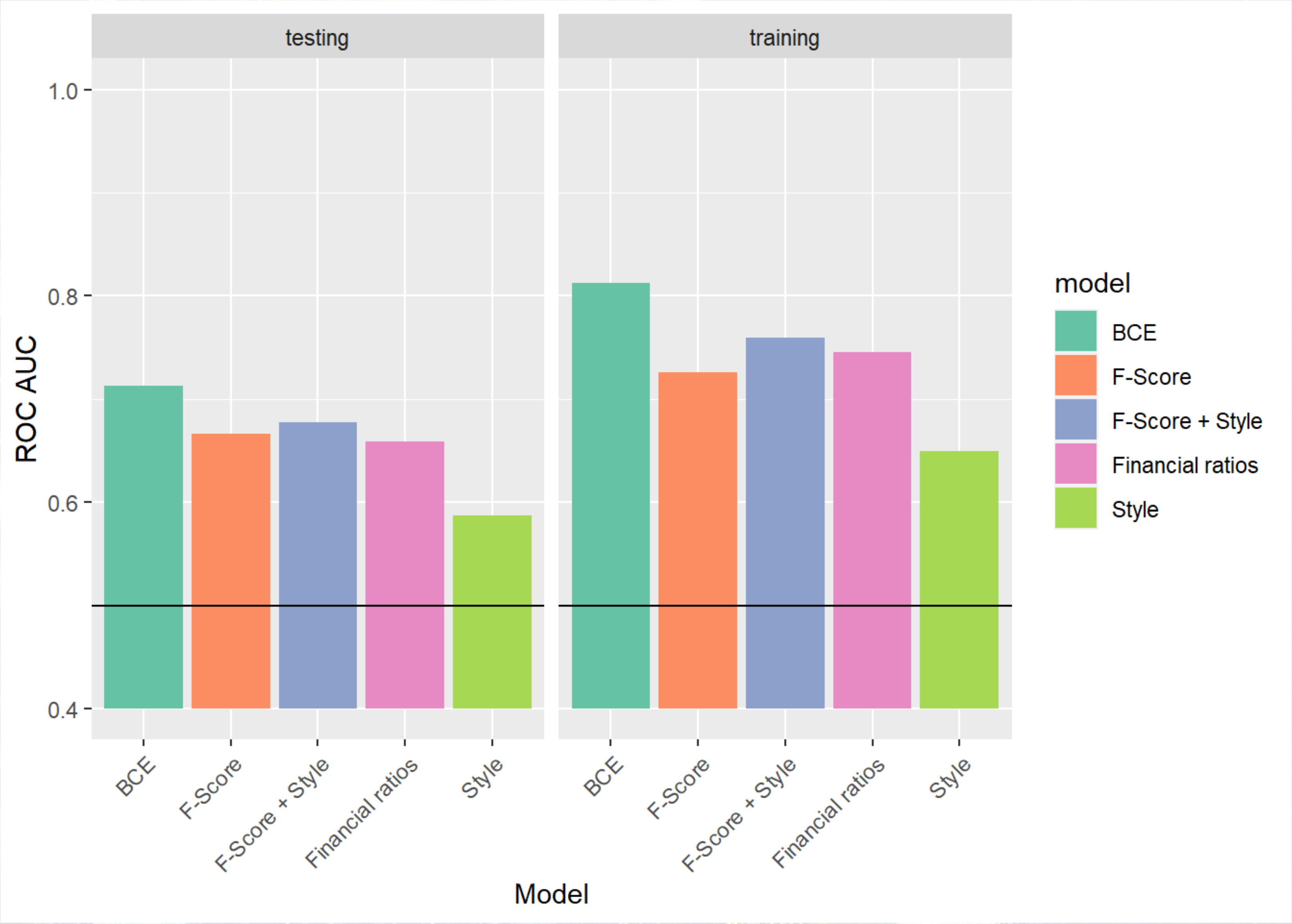
Performance: BCE





Best overall for testing sample so far, but dips in later years

Comparison across all models



The BCE model is superior both on our testing and training data sets.

Econometrics revisited: Machine Learning

Augmenting our statistical analysis

- Traditionally, binary classification problems in statistics are solved using logistic regression
  - This is what we saw so far

Pros of logistic regression

- Regression approaches are familiar
- Easy to run
  - You could even do it in Excel
- Easy to interpret

Cons of logistic regression

- Logistic regression handles *sparse* data poorly
- Ideally you want at least 10% of your data in each group
- Misreporting is [thankfully] sparse!

> If we want a better accuracy, we need to replace logistic regression

- Explore two options: LASSO and XGBoost

What is LASSO?

- Least Absolute Shrinkage and Selection Operator
  - Least absolute: uses an error term like $|\varepsilon|$
  - Shrinkage: it will make coefficients smaller
    - Less sensitive → less overfitting issues
  - Selection: it will completely remove some variables
    - Less variables → less overfitting issues
- Sometimes called L^1 regularization
  - L^1 means 1 dimensional distance, i.e., $|\varepsilon|$

> Great if you have way too many inputs in your model

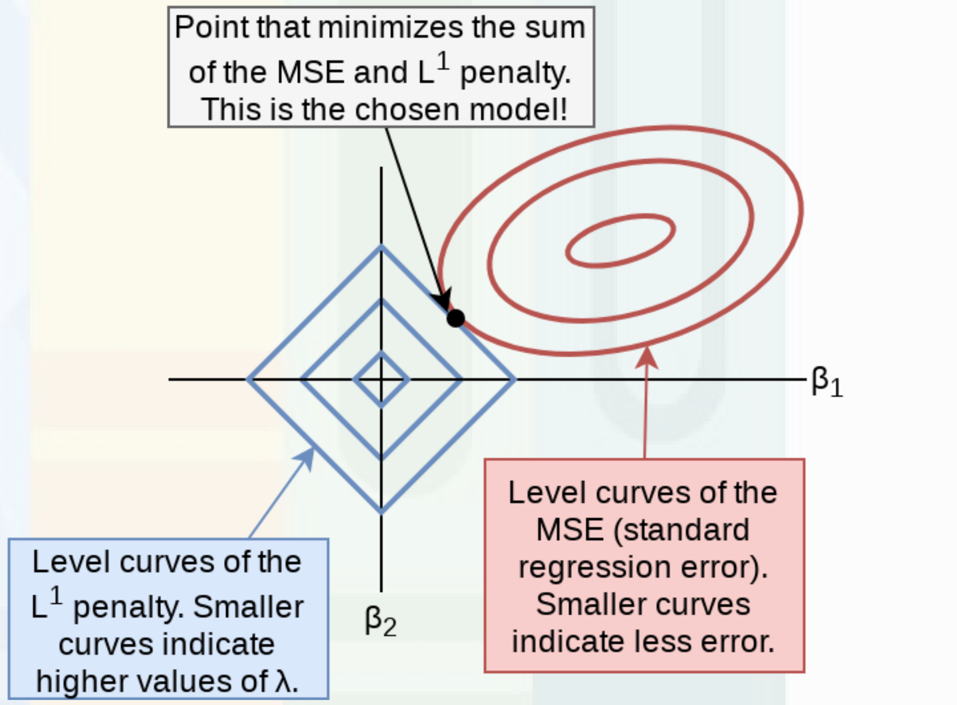
- This is how we can, in theory, put more variables in our model than data points

How does it work?

$$\min_{\beta\in\mathbb{R}} \left\{ \frac{1}{N}|\varepsilon|_2^2 + \lambda|\beta|_1 \right\}$$

- Add an additional penalty term that is increasing in the absolute value of each β
  - Incentivizes lower βs, *shrinking* them
- The selection is part is explainable geometrically

Illustration of LASSO in the *coefficient space* of a regression



Point that minimizes the sum of the MSE and L^1 penalty. This is the chosen model!

Level curves of the L^1 penalty. Smaller curves indicate higher values of λ.

Level curves of the MSE (standard regression error). Smaller curves indicate less error.

Why use it?

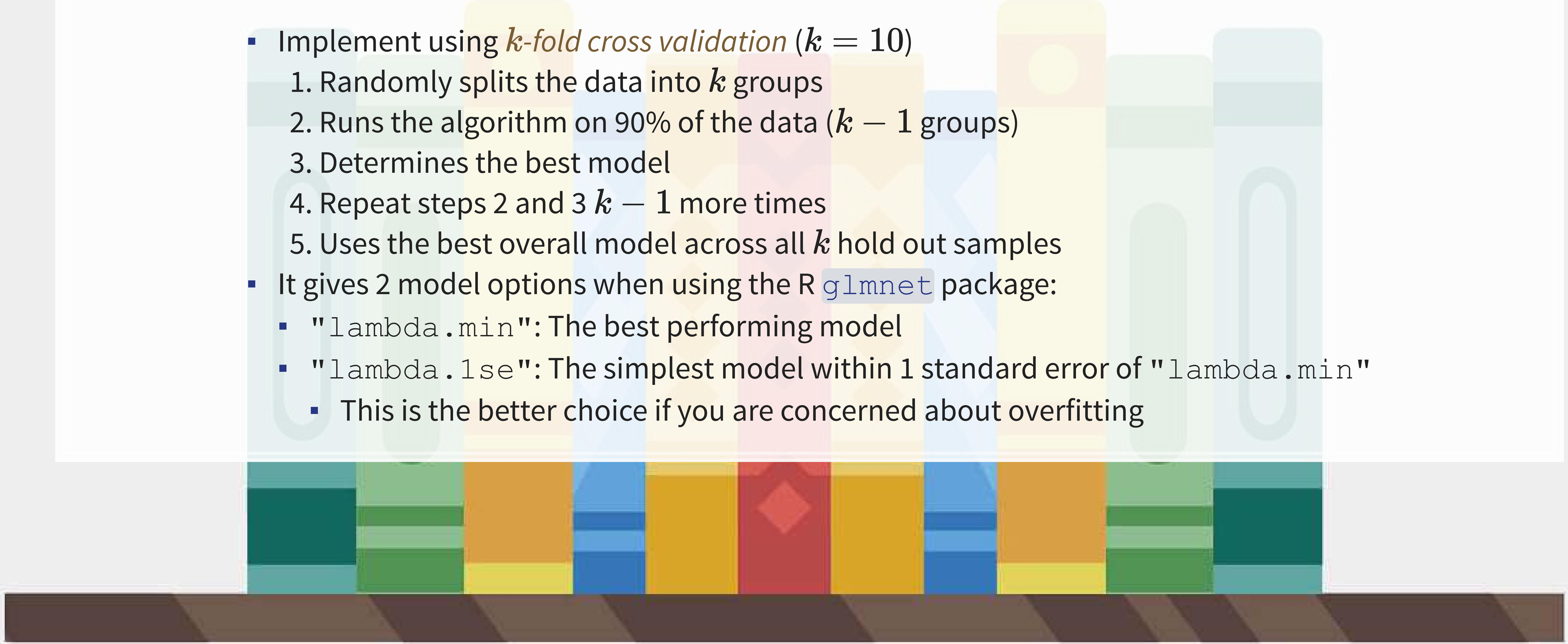
1. We have a preference for simpler models
2. Some problems are naturally very complex
   - Many linkages between different theoretical constructs
3. We don't have a good judgment on what theories are better than others for the problem

> LASSO lets us implement all of our ideas, and then it econometrically kicks out the ineffective ideas (model selection)

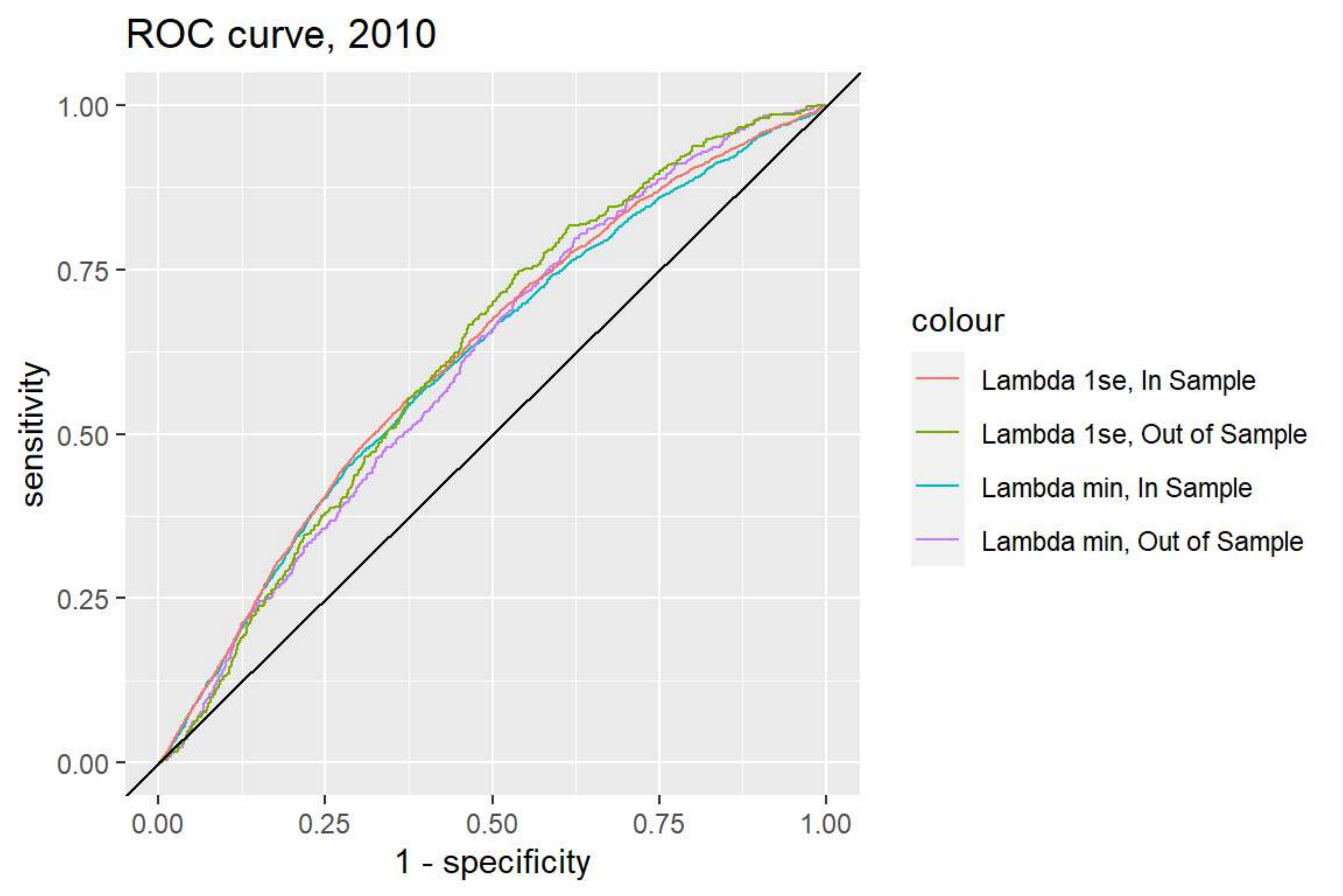
- Drawbacks
  - No p-values on coefficients (can be worked around)
  - Trade off in-sample performance (and sometimes short-run out-of-sample performance) for more stable predictions

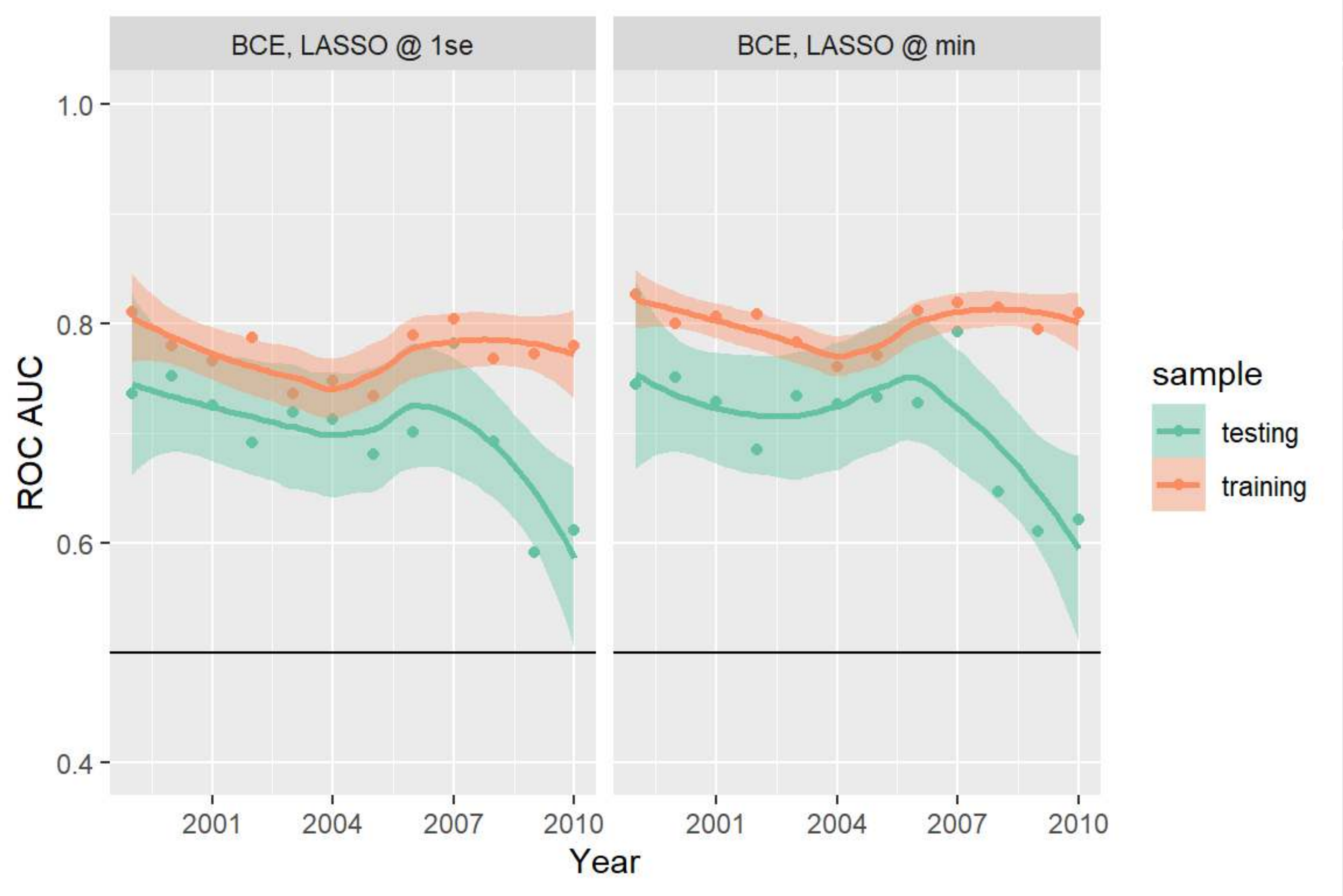
Automating model selection

- Implement using *k-fold cross validation* ($k = 10$)
  1. Randomly splits the data into k groups
  2. Runs the algorithm on 90% of the data ($k - 1$ groups)
  3. Determines the best model
  4. Repeat steps 2 and 3 $k - 1$ more times
  5. Uses the best overall model across all k hold out samples
- It gives 2 model options when using the R `glmnet` package:
  - `"lambda.min"`: The best performing model
  - `"lambda.1se"`: The simplest model within 1 standard error of `"lambda.min"`
    - This is the better choice if you are concerned about overfitting



Performance: BCE with LASSO

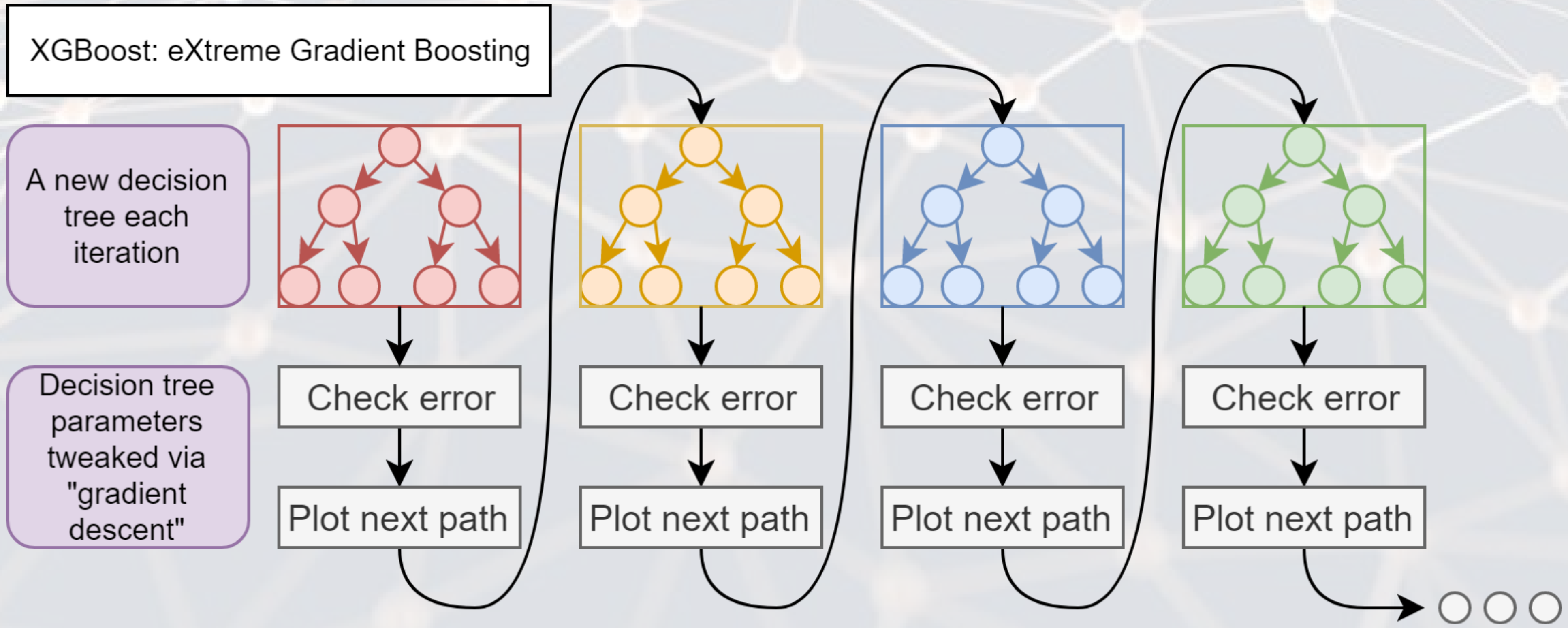




> Performance isn't much different from the BCE model, but the underlying models are much more parsimonious

A nonlinear ML approach: XGBoost

- e**X**treme **G**radient **B**oosting
- A simple explanation:
  1. Start with 1 or more decision trees & check error
  2. Make more decision trees & check error
  3. Use the difference in error to guess a another model
  4. Repeat #2 and #3 until the model's error is stable



XGBoost: eXtreme Gradient Boosting
A new decision tree each iteration
Decision tree parameters tweaked via "gradient descent"
Check error
Plot next path
Check error
Plot next path
Check error
Plot next path
Check error
Plot next path

Benefits of XGBoost

- Tree based
  - Inherently non-parametric (no assumptions on data distribution)
- Non-linear but still somewhat interpretable
- Robust to noise
- Can handle missing or categorical variables (R implementation only)
- Robust to overfitting (somewhat)
- Rooted in AdaBoost (Adaptive Boosting), which uses a sequence of weak learners to build a model
  - Combats overfitting, and the sequence of individually weak models converges to be a strong learner
    - This convergence is mathematically proven!
  - XGBoost isn’t as theoretically founded as Adaboost
    - It trades off some mathematical rigor for flexibility and empirical performance

> As compared to other tree algorithms

- Implements gradient descent to sequentially grow trees
- Parallelizable (so it can be computed efficiently)
- Supports regularization

Drawbacks of XGBoost

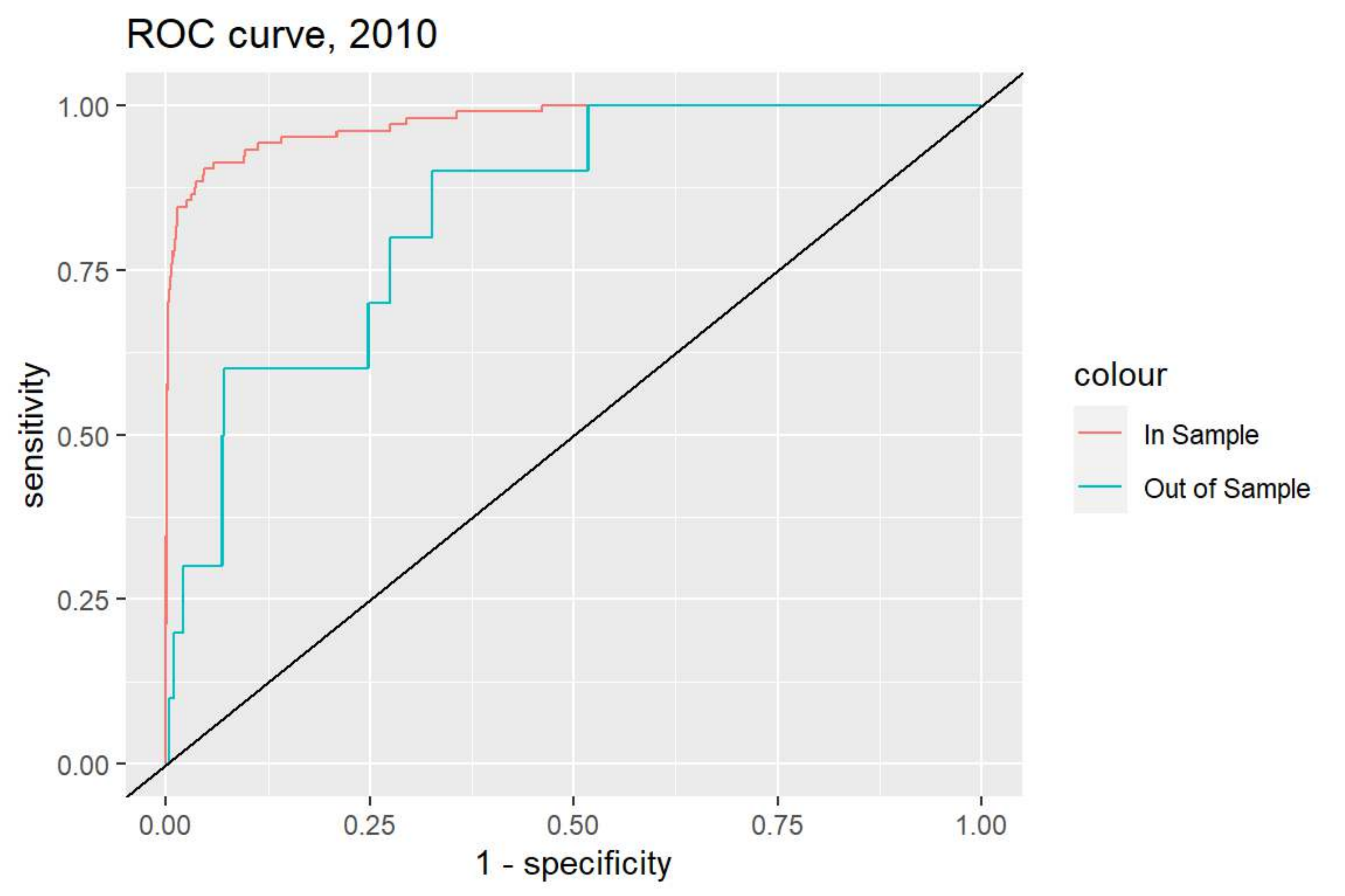
So

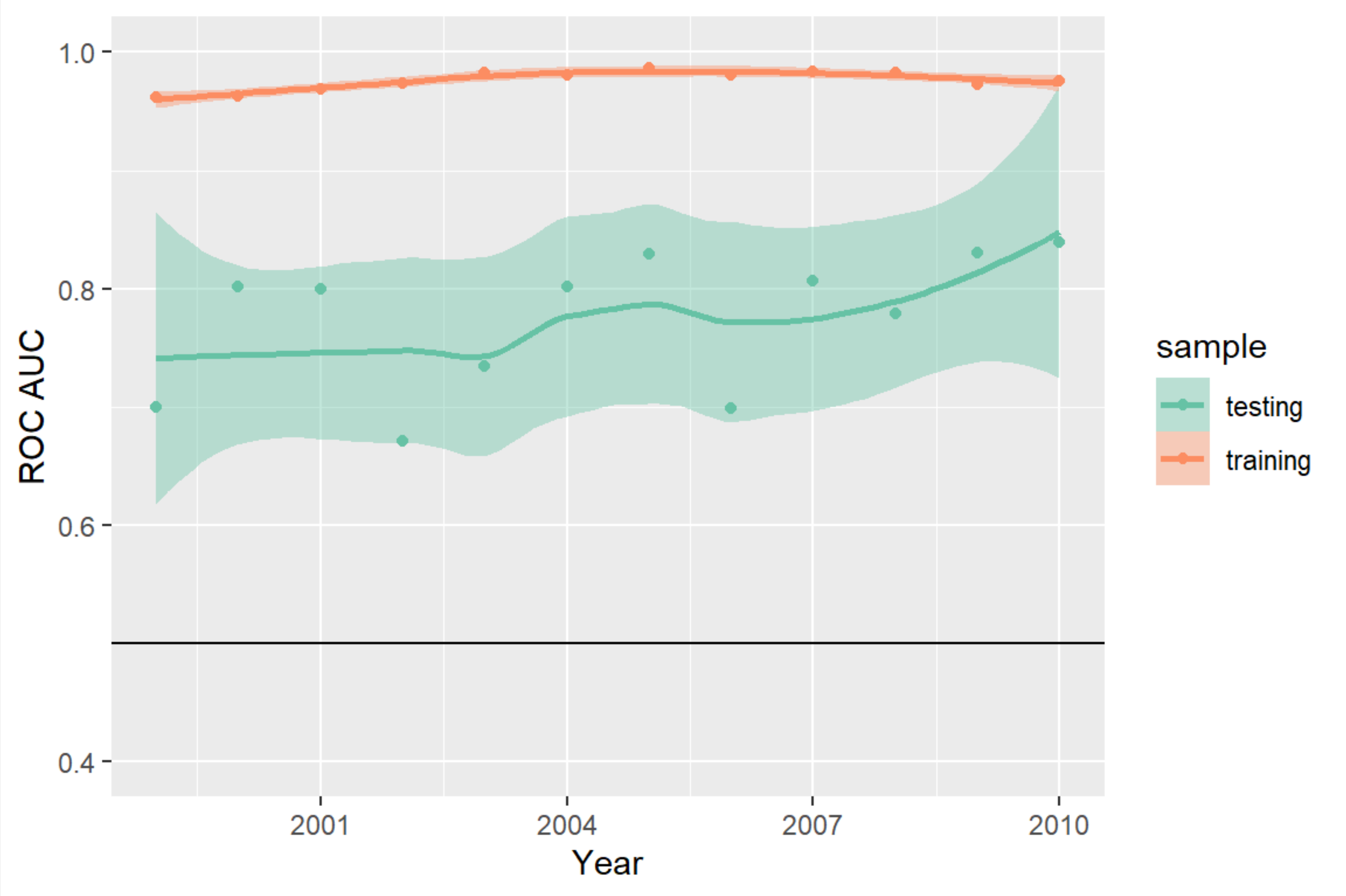
many

hyperparameters.

- This makes it difficult to train a model well
  - But it is hard to beat a well trained XGBoost model with anything else we have discussed thus far
- It may technically be interpretable, but interpreting a big model is still difficult
- Like most tree-based methods, it struggles with extrapolation that is outside the bounds of its input data.

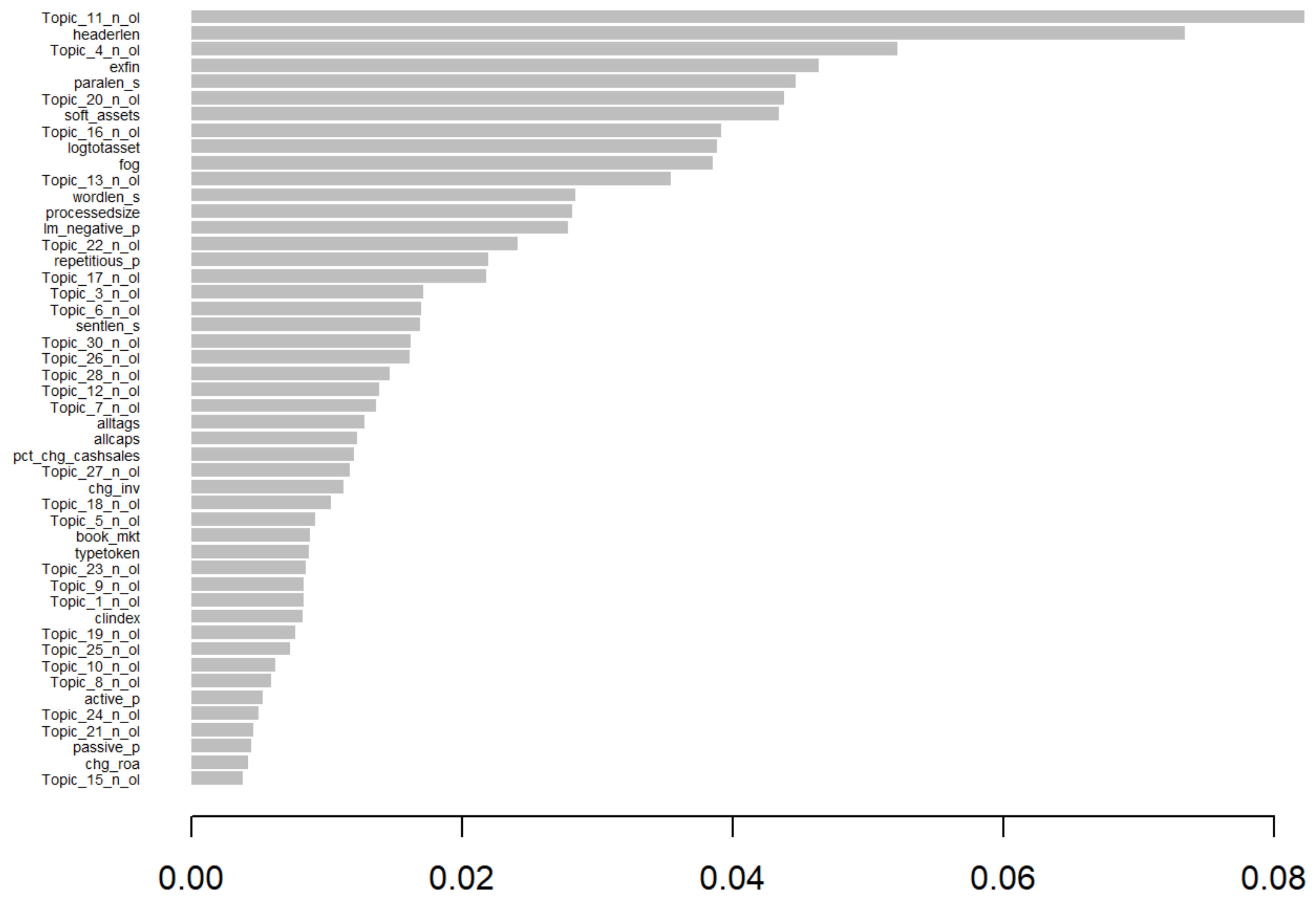
Performance: BCE with XGBoost





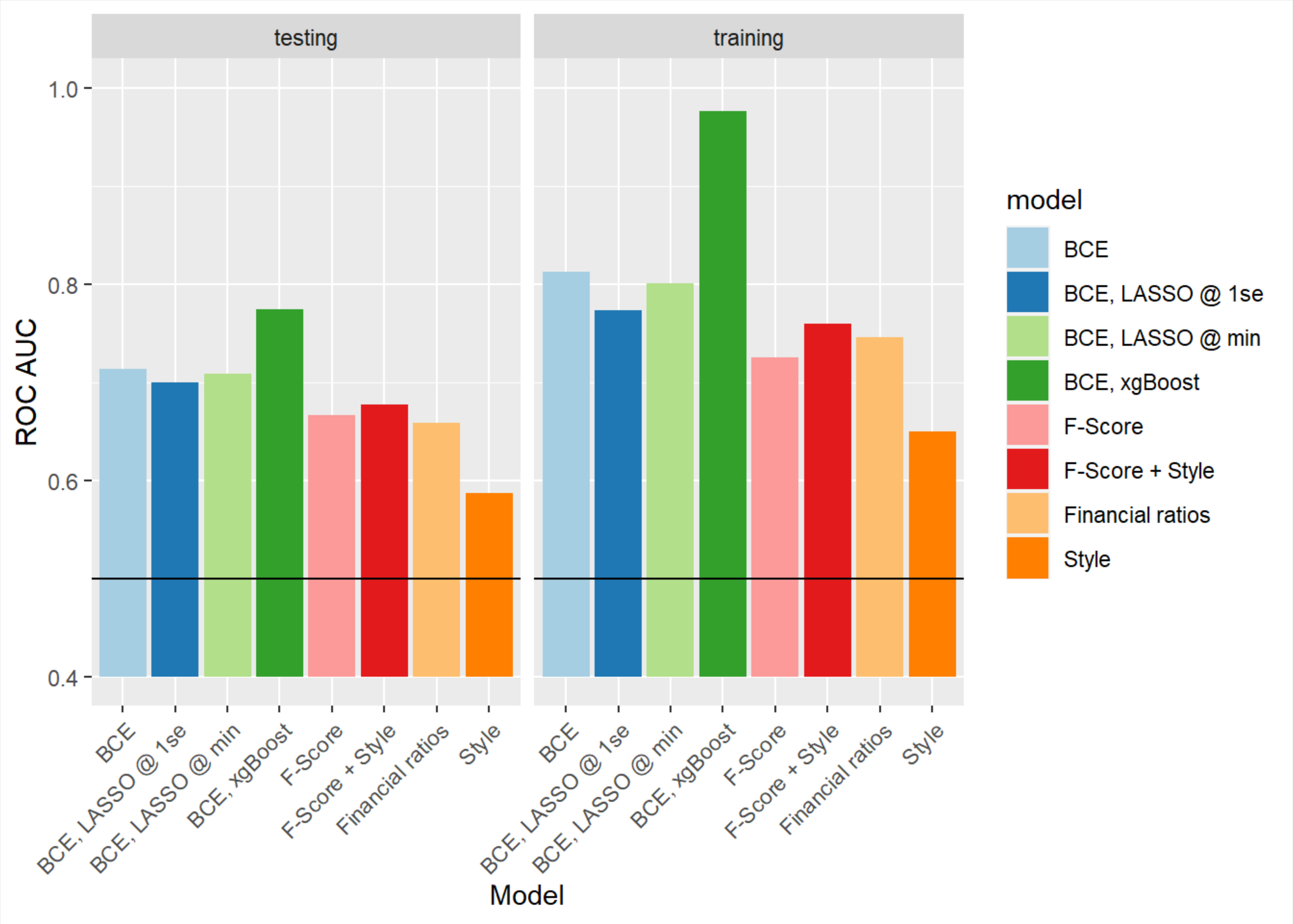
Best overall performance by a wider margin, much more consistent performance

Model explanation



Recap of the models

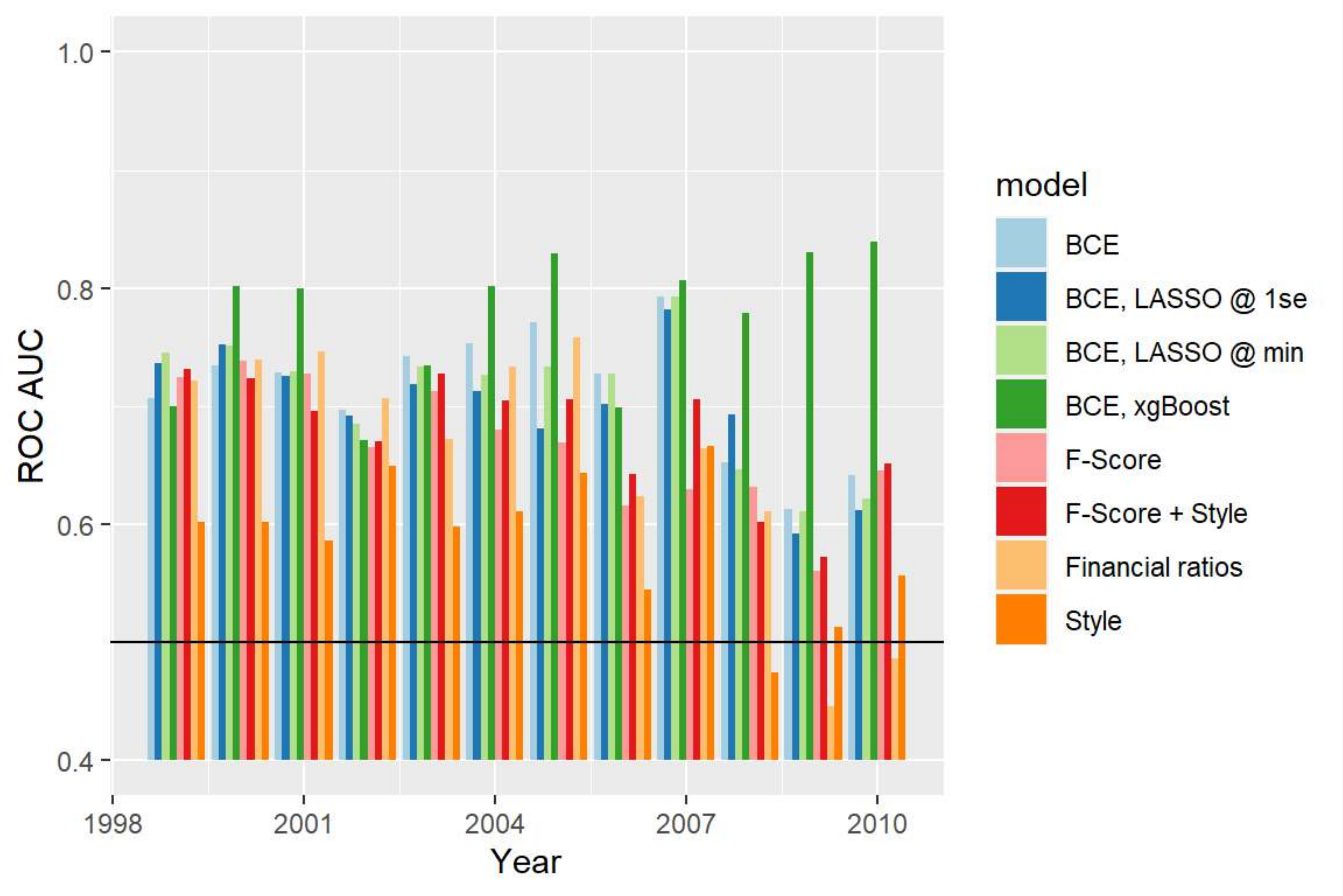
Overall performance

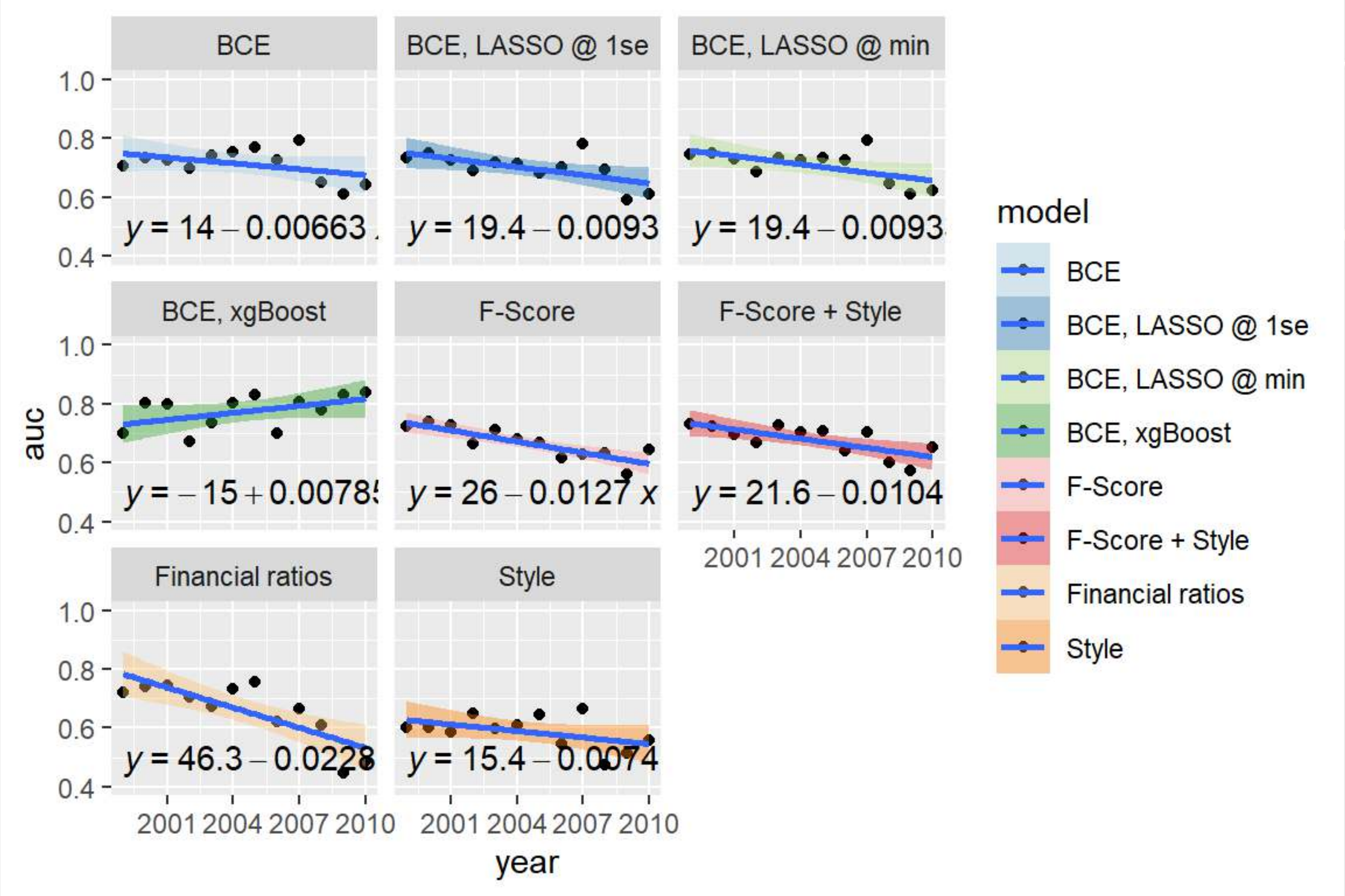


- All BCE models perform well; BCE LASSO @ 1se is parsimonious yet well performing

> BCE with XGBoost is significantly better!

Overall performance by year



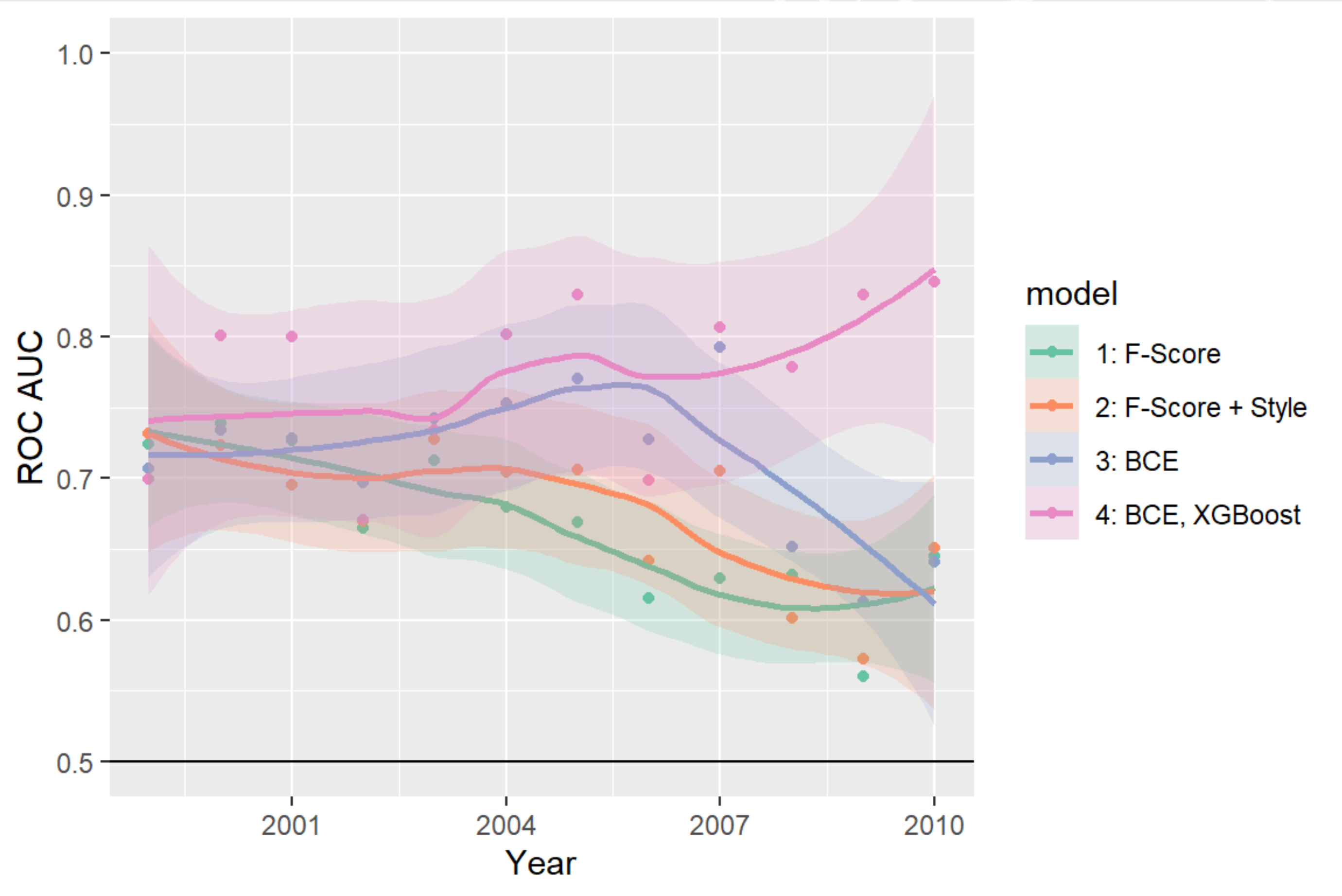


> The BCE model with XGBoost is a clear winner in most years; it is the only model not trending down in performance

End matter

Main takeaways

1. Older models worked well in the past, but start performing poorly by the mid-2000s
2. Including text-based measures helps
3. Including text content helps even more
4. Including non-linear, machine learning based algorithms helps even more



New measures + new techniques ⇒ better performance

Thanks!

Dr. Richard M. Crowley
rcrowley@smu.edu.sg
@prof_rmc
rmc.link/

Packages used for these slides

- `DT`
- `glmnet`
- `ggpmisc`
- `kableExtra`
- `knitr`
- `magrittr`
- `parsnip`
- `plotly`
- `recipes`
- `revealjs`
- `scales`
- `tidyverse`
- `xgboost`
- `yardstick`

Appendix

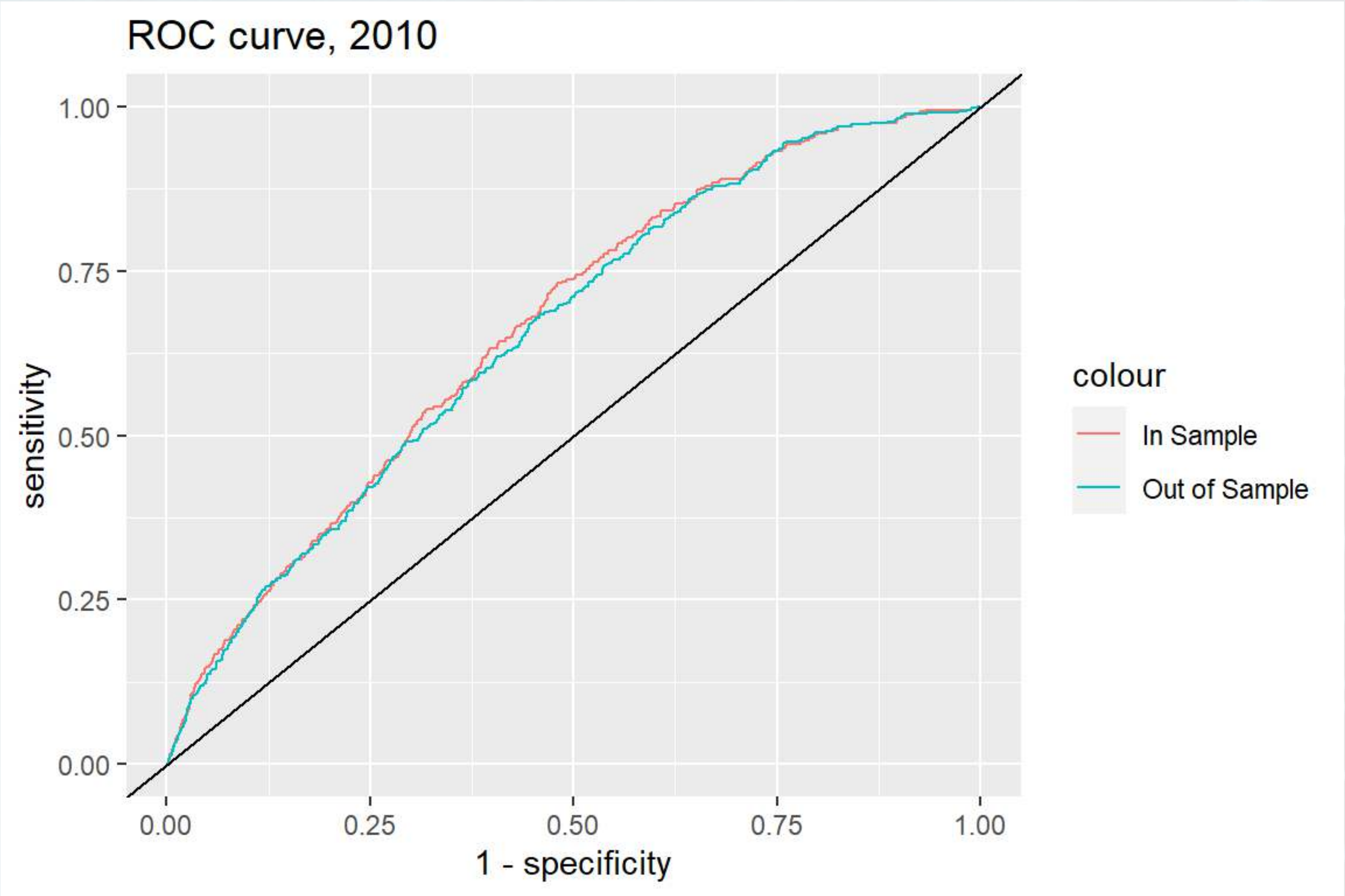
Comparing multiple models

- Performance measures:
  1. **ROC AUC**
  2. Fisher statistics
  3. Performance at a reasonable cutoff (5%)
  4. `NDCG@k` (usually used in ranking problems)

> ROC AUC and Fisher statistics also allow us to statistically compare across models

ROC AUC for windowed approaches

- **R**eceiver **O**perator **C**urve
  - ROC curve compares sensitivity and specificity of a model
    - Sensitivity: True positive rate
    - Specificity: True negative rate
- A better measure has a curve closer to the upper left corner
- A random measure has a curve at a 45-degree line



> **A**rea **U**nder the **C**urve (AUC): What is the probability that a randomly selected `AAER=1` is ranked higher than a randomly selected `AAER=0`? A good score is above 0.70

Comparing with ROC AUC

- Can aggregate ROC AUCs via pooling predictions together
  - With clustering by year
- Higher aggregate AUC is better, but direct comparison is tricky
- Bootstrapping allows for generating test statistics for ROC AUCs, which can be compared with a Wald test
  - Available in Stata as part of `rocreg`

Comparing with Fisher Statistics

- Fisher (1934) provides a solution to aggregating p-values into a X^2 test statistic

$$-2\sum_{i=1}^{k}\log\left(p\text{-}value_i\right) \sim X_{2k}^2$$

- The difference of X^2 distributed variables follows a Variance Gamma distribution
- For 2 Fisher statistics X_1 and X_2 each with k observations:

$$\mathbb{P}\left(X_1 < X_2\right) = \int_{-\infty}^{X_1-X_2} \frac{1}{2^k\sqrt{\pi}\Gamma(k)}|z|^{k-1/2}K_{k-1/2}(|z|)dz$$

- where Γ is the gamma function and $K_{k-1/2}$ is the modified Bessel function of the second kind

Other methods of measuring performance

NDCG @k: **N**ormalized **D**iscounted **C**umulative **G**ain @k

- Measures *ranking* quality, used for search engine optimization
- k is a specified percentile or # of observations
- “DCG” measures the # of true positives in k of the prediction score
- “N” is to divide the DCG by the theoretically optimal DCG to normalize to a [0,1] interval

Counts at different thresholds

- E.g., at a 95% cutoff, the BCE model captures 96 AAERS, whereas traditional models only capture 70
- Easy to interpret economically
- Maps well to what regulators do in practice